

4/5/1 (Item 1 from file: 347)  
DIALOG(R) File 347:JAPIO  
(c) 2004 JPO & JAPIO. All rts. reserv.

04800538 \*\*Image available\*\*  
ONE-CHIP MICROCOMPUTER SYSTEM

PUB. NO.: 07-093138 [ JP 7093138 A]  
PUBLISHED: April 07, 1995 (19950407)  
INVENTOR(s): MIYAZAWA AZUMA  
ITO JUNICHI  
IMAI YUJI  
OKUMURA YOICHIRO  
HARA NOBORU  
KOBAYASHI KAZUTADA  
FUJIBAYASHI KENJI  
SAITO YUICHI  
KAWAMURA SHOJI

APPLICANT(s): OLYMPUS OPTICAL CO LTD [000037] (A Japanese Company or Corporation), JP (Japan)

APPL. NO.: 06-037150 [JP 9437150]

FILED: March 08, 1994 (19940308)

INTL CLASS: [6] G06F-009/06; G06F-011/28; G06F-015/78

JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units);  
45.4 (INFORMATION PROCESSING -- Computer Applications)

JAPIO KEYWORD: R131 (INFORMATION PROCESSING -- Microcomputers & Microprocessors)

#### ABSTRACT

PURPOSE: To provide a one-chip microcomputer (CPU) system automatically discriminating whether a mask ROM built in a CPU is before or after correction.

CONSTITUTION: An AF circuit 12, AE circuit 13, EEPROM 14 for storing the adjusted value of a camera or data for ROM correction, connector 15 for external communication to connect an external device composed of the adjuster of the camera or a device for writing data for ROM connection and stroboscope 16 or the like are connected to a CPU 11 which stores the version in a built-in mask ROM 11a and can correct programs. Further, a power switch 18, rear lid switch 19, rewind switch 20, 1st release switch 21, 2nd release switch 22, zoom-up switch 23, zoom-down switch 24, mode switch 25 and stroboscopic mode switch 26 or the like are connected to the CPU 11. The version of the mask ROM 11a is read from the outside, and correcting data are selected.

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-93138

(43) 公開日 平成7年(1995)4月7日

(51) Int.Cl.<sup>6</sup>

G 0 6 F 9/06  
11/28  
15/78

識別記号

3 3 0 C  
5 1 0 K

庁内整理番号

9367-5B  
9290-5B

F I

技術表示箇所

審査請求 未請求 請求項の数 3 O L (全 26 頁)

(21) 出願番号 特願平6-37150

(22) 出願日 平成6年(1994)3月8日

(31) 優先権主張番号 特願平5-51752

(32) 優先日 平5(1993)3月12日

(33) 優先権主張国 日本 (J P)

(71) 出願人 000000376

オリンパス光学工業株式会社  
東京都渋谷区幡ヶ谷2丁目43番2号

(72) 発明者 宮沢 東

東京都渋谷区幡ヶ谷2丁目43番2号 オリ  
ンパス光学工業株式会社内

(72) 発明者 伊藤 順一

東京都渋谷区幡ヶ谷2丁目43番2号 オリ  
ンパス光学工業株式会社内

(72) 発明者 今井 右二

東京都渋谷区幡ヶ谷2丁目43番2号 オリ  
ンパス光学工業株式会社内

(74) 代理人 弁理士 鈴江 武彦

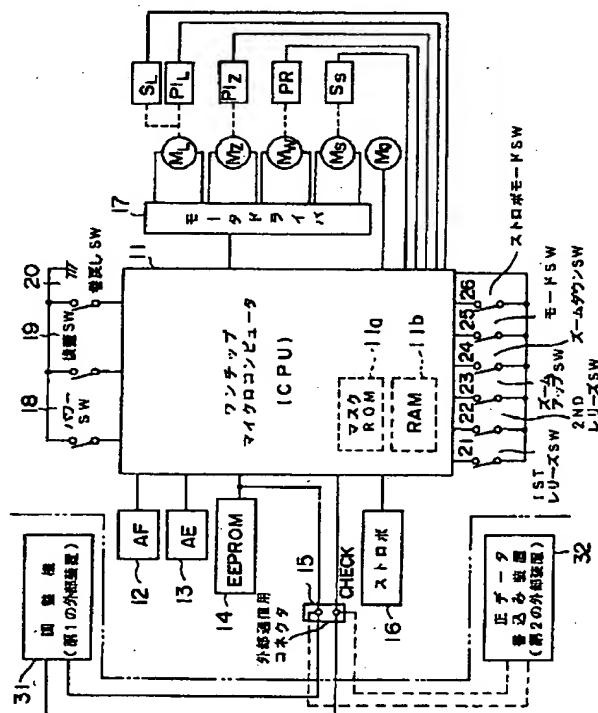
最終頁に続く

(54) 【発明の名称】 ワンチップマイクロコンピュータシステム

(57) 【要約】

【目的】 マイクロコンピュータに内蔵されるマスクROMの修正前後を自動的に判別するワンチップマイクロコンピュータ (CPU) システムを提供する。

【構成】 内蔵されるマスクROMにそのバージョンが記憶されたプログラム修正可能なCPU11には、AF回路12と、AE回路13と、カメラの調整値やROM修正用データを記憶するEEPROM14と、カメラの調整機やROM修正用データを書込む装置からなる外部装置を接続する外部通信コネクタ15と、ストロボ16等が接続される。また、上記CPU11には、パワースイッチ18、後蓋スイッチ19、巻戻しスイッチ20、1stレリーズスイッチ21、2ndレリーズスイッチ22、ズームアップスイッチ23、ズームダウンスイッチ24、モードスイッチ25及びストロボモードスイッチ26等が接続される。前記マスクROMのバージョンを外部から読出して修正データを選択する。



## 1

## 【特許請求の範囲】

【請求項1】 マスクROMに記憶されたプログラム内容を擬似的に変更可能な機能を有すると共に、前記マスクROMの種類を示すROMバージョンデータを記憶しているワンチップマイクロコンピュータと、前記マスクROMに記憶されたプログラム内容を擬似的に修正するための修正データを記憶可能であると共に、前記ROMバージョンデータに対応した修正データを記憶可能な記憶領域を有する電気的に書き換え可能な不揮発性メモリと、前記ワンチップマイクロコンピュータもしくは前記不揮発性メモリに前記修正データを入力するための接続端子とを具備することを特徴とするワンチップマイクロコンピュータシステム。

【請求項2】 前記システムは前記接続端子を介して外部装置に接続可能であり、この外部装置は複数の修正データを記憶し、前記ROMバージョンデータに基づいて前記複数の修正データから適切な修正データを選択して前記システムに入力することを特徴とする請求項1に記載のワンチップマイクロコンピュータシステム。

【請求項3】 ワンチップマイクロコンピュータ本体と、修正データを受信するためのデータ入力手段と、ROMバージョンと前記ワンチップマイクロコンピュータ本体による制御のためのプログラム内容とを記憶したマスクROMと、前記マスクROMに記憶されたプログラム内容を前記データ入力手段を介して受信した修正データに基づいて擬似的に変更する擬似的変更手段とを含んでなるワンチップマイクロコンピュータと、前記ワンチップマイクロコンピュータの外部に配置されるもので、前記マスクROMに記憶されたプログラム内容を変更するための前記修正データが記憶されていると共に、さらに前記マスクROMに記憶されたROMバージョンに適合する修正データが書き込まれる不揮発性メモリとを具備することを特徴とするワンチップマイクロコンピュータシステム。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 この発明は、マスクROMに書込まれたプログラム内容を擬似的に修正可能なワンチップマイクロコンピュータシステムに関するものである。

## 【0002】

【従来の技術】 近年のカメラは、ワンチップマイクロコンピュータ（ワンチップCPU）に内蔵されたマスクタイプのリードオンリメモリ（マスクROM）に書込まれたプログラムに従って動作するように構成されている。

【0003】 このようなプログラムは、カメラのシステム設計時に考えられる各種のアクセサリとの組合せに於いて、最適に動作するように作成されている。また、このようなカメラシステムは通常、パワーオンリセットでマスクROMの初期アドレスから順次プログラムを処

## 2

理して、最終的には低消費電力モードで停止する。

【0004】 この一連の処理は、外部から希望のアドレスで停止させたり、処理の進捗状況すなわちワンチップCPUに内蔵されるプログラムカウンタの値やランダムアクセスメモリ（RAM）の値等を外部からチェックすることはできない。

## 【0005】

【発明が解決しようとする課題】 ところで、上述したワンチップCPUを有する従来のカメラシステムに於いて、カメラのプログラムは、システム設計時に考えられる全てのアクセサリに対応したプログラムを有している。

【0006】 しかしながら、カメラシステムとしての製品の生産過程に於いて、例えばカメラシステム設計時に考えられなかった不具合の対策等が施されると、それに伴ってワンチップCPU等を搭載する電気回路基板等、他の部品の仕様変更が生じてくる。

【0007】 このような場合、従来は、その仕様変更による部品のバージョンが改定できないうちは、最初に使

用されている部品を使用して対策を施しているため、部品を目視しただけでは修正前と修正後の何れであるかを判別することが困難であった。

【0008】 したがって、例えば生産ラインに於いて対策を施す必要があるものであるか否かが分からなくなるという虞れがあった。この場合、特開平3-186927号公報に開示されているプログラム修正可能なマイクロコンピュータを搭載したシステムのプログラム修正に関する技術によって、プログラム修正自体はマイクロコンピュータによって行うことができるので、上述のような不具合の対策を行うことができる。

【0009】 しかしながら、このようなプログラム修正によるものでも、マイクロコンピュータのROMが修正された後であっても、修正前と修正後のマイクロコンピュータの区別をする方法については記載されていないものであった。

【0010】 したがって、生産ラインやリペア店では、その製品が不具合対策が施されたものであるか否かということ、混乱して判別しにくく、管理がしにくいという課題があった。

【0011】 この発明は、このような課題に着目してなされたもので、その目的とするところは、内蔵されるマスクROMが修正された後のマイクロコンピュータと修正前のマイクロコンピュータとを自動的に判別し、生産ラインやリペア店で混乱を生じることなく、管理が容易なワンチップマイクロコンピュータシステムを提供することにある。

## 【0012】

【課題を解決するための手段】 この発明によると、マスクROMに記憶されたプログラム内容を擬似的に変更可能な機能を有すると共に、前記マスクROMの種類を示

## 3

すROMバージョンデータを記憶しているワンチップマイクロコンピュータと、前記マスクROMに記憶されたプログラム内容を擬似的に修正するための修正データを記憶可能であると共に、前記ROMバージョンデータに対応した修正データを記憶可能な記憶領域を有する電氣的に書き換え可能な不揮発性メモリと、前記ワンチップマイクロコンピュータもしくは前記不揮発性メモリに前記修正データを入力するための接続端子とを具備することを特徴とするワンチップマイクロコンピュータシステムが提供される。

【0013】また、この発明によると、前記システムは前記接続端子を介して外部装置に接続可能であり、この外部装置は複数の修正データを記憶し、前記ROMバージョンデータに基づいて前記複数の修正データから適切な修正データを選択して前記システムに入力することを特徴とするワンチップマイクロコンピュータシステムが提供される。

【0014】また、この発明によると、ワンチップマイクロコンピュータ本体と、修正データを受信するためのデータ入力手段と、ROMバージョンと前記ワンチップマイクロコンピュータ本体による制御のためのプログラム内容とを記憶したマスクROMと、前記マスクROMに記憶されたプログラム内容を前記データ入力手段を介して受信した修正データに基づいて擬似的に変更する擬似的変更手段とを含んでなるワンチップマイクロコンピュータと、前記ワンチップマイクロコンピュータの外部に配置されるもので、前記マスクROMに記憶されたプログラム内容を変更するための前記修正データが記憶されていると共に、さらに前記マスクROMに記憶されたROMバージョンに適合する修正データが書き込まれる不揮発性メモリとを具備することを特徴とするワンチップマイクロコンピュータシステムが提供される。

【0015】

【作用】この発明にあって、ワンチップマイクロコンピュータは、マスクROMに記憶されたプログラム内容を擬似的に変更する擬似的変更手段を有し、このワンチップマイクロコンピュータの外部に配置される不揮発性メモリは、上記プログラムの内容を修正するための修正データを記憶していると共に、データ入力手段によって、外部から上記不揮発性メモリにデータが書き込まれ、制御対象が上記ワンチップマイクロコンピュータによって制御される。

【0016】また、上記マスクROMはROMバージョンを記憶し、上記不揮発性メモリは基板バージョンを記憶する。そして、上記データ入力手段を介して、上記データの書き込みにあたって、上記ROMバージョンと上記基板バージョンを基に、上記修正データを選択して上記不揮発性メモリに書き込む。

【0017】

【実施例】以下、図面を参照してこの発明の実施例を説

## 4

明する。図1は、この発明のワンチップマイクロコンピュータシステムが適用されたカメラシステムを例にとったブロック構成図である。

【0018】同図に於いて、11はこのカメラシステムのシーケンス制御を行うためのワンチップマイクロコンピュータ（以下CPUと記す）であり、制御用のプログラムデータを格納する内蔵マスクROM（以下マスクROMと記す）11aの内容を疑似的に変更できる機能を有している。

10 【0019】このCPU11には、被写体までの距離を測定する自動焦点（AF）回路12と、被写体の明るさを測定する自動露光（AE）回路13と、カメラシステムの調整値やROM修正用データを記憶するもので電氣的に書換可能な不揮発性メモリ（EEPROM）14と、カメラの調整機（第1の外部装置）31や、ROM修正用データを書込む修正データ書き込み装置（第2の外部装置）32を接続するための外部通信用コネクタ15と、ストロボ充電や発光を行うストロボ16とが接続されている。

20 【0020】ここで、外部通信用コネクタ15はCPU11を搭載する電気回路（プリント配線）基板（図示せず）上に形成された導電体パターンでもよい。また、第1及び第2の外部装置31、32は、カメラの調整時や後述するROM修正データを書き込むときのみ外部通信用コネクタ15に接続される。

【0021】ここで、第1の外部装置31は工場におけるカメラの組み立て、調整時に使用される。また、第2の外部装置32はリペア店でのカメラ修理時における後述するデリバティモードの書き込み時に使用される。

30 【0022】なお、図1ではEEPROM14はCPU11の外部に位置しているが、EEPROM内蔵のCPUを使用してもよいことは言うまでもない。上記CPU11には、各種モータ駆動用のモータドライバ17を介して、フォーカスレンズを駆動するモータM<sub>L</sub>、ズームレンズを駆動するモータM<sub>Z</sub>、フィルムの巻上げ巻戻しを行うモータM<sub>W</sub>、シャッタを駆動するモータM<sub>S</sub>と、上記シャッタを閉じるためのマグネットM<sub>G</sub>とが接続されている。

40 【0023】また、CPU11には、フォーカスレンズの初期位置を検出するスイッチS<sub>L</sub>と、フォーカスレンズの単位駆動量（位置）を検出するフォトインタラプタP<sub>I<sub>L</sub></sub>と、ズームレンズの位置を検出するフォトインタラプタP<sub>I<sub>Z</sub></sub>と、フィルムのパーフォレーションを検出するフォトリフレクタP<sub>R</sub>と、シャッタの初期位置を検出するスイッチS<sub>S</sub>が接続されている。

50 【0024】さらに、上記CPU11には、パワースイッチ18、後蓋スイッチ19、巻戻しスイッチ20、ファースト（1st）リリーススイッチ21、セカンド（2nd）リリーススイッチ22、ズームアップスイッチ23、ズームダウンスイッチ24、モードスイッチ2

5及びストロボモードスイッチ26の各種スイッチが接続されている。

【0025】なお、1stリリーススイッチ21と2ndリリーススイッチ22は2段スイッチを構成しており、1段目で1stリリーススイッチ21、2段目で2ndリリーススイッチ22がオンになるスイッチである。

【0026】図2は、このカメラシステムの全体の動作を説明するメインフローチャートである。同図を参照すると、先ず、図示されない電池がカメラ本体に装填されると、パワーオンリセットがかかる。

【0027】ここで、CPU11のスタックポイントを設定してから(ステップS1)、ポートやレジスタの初期設定を行う(ステップS2)。次に、「ROM修正データ設定」のサブルーチンにより、ROM修正データが必要な場合は、EEPROM14にデータを設定する(ステップS3)。

【0028】その後、割込み設定を行い、必要な割り込みを許可する(ステップS4)。次いで、バッテリーチェックを行い(ステップS5)、その結果が良ければカメラシステムのシーケンス制御に移行する。

【0029】このカメラシステムのシーケンス制御では、先ずパワースイッチ18の状態をチェックする(ステップS6)。ここで、パワースイッチ18がオフであれば、図示しない液晶表示器(LCD)の表示を消し(ステップS7)、ポート等に電流が流れない状態にして消エネモードにし(ステップS8)、CPU11をスタンバイにする。

【0030】一方、上記ステップS6において、パワースイッチ18がオンであれば、LCDの表示を行い(ステップS9)、続いてストロボ充電を行う(ステップS10)。

【0031】次に、各スイッチの状態を判定する。すなわち、先ず1stリリーススイッチ21が押されたか否かを判定し(ステップS11)、このスイッチが押されていれば「リリース処理」を実行する(ステップS12)。

【0032】次いで、ズームスイッチを判定し(ステップS13)、ズームアップスイッチ23またはズームダウンスイッチ24が押されていれば「ズーム処理」を実行する(ステップS14)。

【0033】次に、ストロボモードスイッチ26を判定して(ステップS15)、このスイッチが押されていれば「ストロボモード処理」を実行する(ステップS16)。さらに、モードスイッチ25が押されていれば(ステップS17)、「モード処理」を行う(ステップS18)。

【0034】このようにして、パワースイッチ18がオフになるまで、以上のステップS6～S18の動作を繰返す。次に、図3を参照して、サブルーチン「リリース

処理」の動作を説明する。

【0035】先ず、AF回路12にて測距を行った後(ステップS21)、AE回路13で測光を行う(ステップS22)。そして、2ndリリーススイッチ22がオフであれば、オンになるまで待機し(ステップS23)、1stリリーススイッチ21がオフになっていれば(ステップS24)、「リリース処理」を中断してメインフローに戻る。

【0036】上記ステップS23に於いて、2ndリリーススイッチ22がオンになると、フォーカスレンズを駆動し(ステップS25)、次いで、ピントを合せてシャッターを開き(ステップS26)、フィルムを巻上げた後(ステップS27)、このリリース処理を終る。

【0037】図4は、割込み処理の動作を示すフローチャートである。図4(a)に示される第1の割込み処理は、パワースイッチ18がオフで、且つCPU11がスタンバイ状態にあるときに、パワースイッチ18がオンになると割込みが発生する。

【0038】すると、スタンバイを解除して、図2のメインフローチャートのステップS1から処理を実行する。図4(b)は、第2の割込み処理としての、後蓋開閉スイッチ19の割込み処理のフローチャートである。

【0039】先ず、後蓋スイッチ19の状態を判定し(ステップS31)により、カメラの後蓋(図示せず)が開になった場合は、LCDの駒数をリセットする(ステップS32)。

【0040】一方、後蓋が閉になった場合は、図示しないフィルムパトローネの有無を検出する(ステップS33)。ここで、フィルムパトローネが有る場合は「オートロード」により、フィルムを空送りする(ステップS34)。

【0041】これに対して、フィルムパトローネが無い場合は「オートロード」は行わない(ステップS34)。図4(c)は、第3の割込み処理としての巻戻しスイッチ20の割込み処理を示す。

【0042】巻戻しスイッチ20が押された場合は、フィルムを巻戻す(ステップS35)。また、図4(d)は、第4の割込み処理としての、外部通信用コネクタ15のCHECK(チェック)端子が“L(ローレベル)”になった場合の割込みを示す。

【0043】CHECK端子は、通常CPU11の内部プルアップ抵抗で“H(ハイレベル)”になっているので、外部から強制的に“L”にした場合のみ割り込みが発生する。

【0044】すなわち、CHECK端子が“L”になった場合は、「チェッカ通信」を行う(ステップS36)。図5は、この「チェッカ通信」の動作を説明するサブルーチンである。

【0045】なお、このチェッカ通信の詳細は、本出願人による特開平2-941号公報に記載されているの

で、その詳細な説明は省略するものとし、ここでは概念のみ説明する。

【0046】まず、CPU11から、外部に対して同期信号及びシリアル通信用のクロックを出力し（ステップS41、S42）、外部よりデータを受取る（ステップS43）。

【0047】ここで、データが無ければ終了であるが、データを受取った場合は、メモリのモードを判定した後（ステップS44）、リードモードであればそのデータの意味を解説し、CPU11内の指定されたアドレスのデータを読む（ステップS45）。 10

【0048】その後、シリアルラインにデータを出力して（ステップS46）、本ルーチンを抜ける。上記メモリのモードがライトモードであった場合（ステップS47）は、確認データをチェック（ステップS48）し、その結果が良ければ、指定されたアドレスにデータを書込む（ステップS49）。

【0049】また、サブルーチンコールモードであった場合（ステップS50）は、確認データをチェックし（ステップS48）、その結果が良ければ、指定されたアドレスのサブルーチンをコールして（ステップS52）、指定されたアドレスのサブルーチンを実行する。 20

【0050】さらに、外部との通信のみを行う連続通信モードであった場合（ステップS53）は、続いて連続通信モード中で有るかを判定する（ステップS54）。そして、連続通信モード中でない場合は、再びチェッカ通信を行った後（ステップS55）、CHECK端子が“H”になるまで「チェッカ通信」を行う（ステップS56）。

【0051】そして、連続通信を中止するオフモードであった場合は（ステップS57）、連続モード中であるかを判定（ステップS58）し、連続モード中であれば、スタックポインタを1レベル戻す（ステップS59）。 30

【0052】次に、図6のサブルーチンを参照して、「ストロボモード処理」について説明する。まず、ストロボモードスイッチ26が押される毎に、ストロボモードスイッチ26用としてCPU11に内蔵されたRAM11b中のストロボモードRAMをインクリメントする（ステップS61）。

【0053】次いで、ストロボモードRAMの値を判定する（ステップS62、S63）。ここで、ストロボモードRAMの値が“0”のときはオートストロボモードに設定する（ステップS64）し、ストロボモードRAMが“1”のときは、ストロボ発光禁止モードに設定する（ステップS65）。 40

【0054】さらに、ストロボモードRAMが“2”のときは、ストロボ常時発光モードに設定する（ステップS64）。上記ストロボモードRAMは“2”までしか使用していないので、ストロボモードRAMが“3以

上”になったときは、それを“0”にクリア（ステップS66）して、オートストロボモードに設定する（ステップS67）。

【0055】図7は、「モード処理」の動作を説明するサブルーチンである。モードスイッチ25が押される毎に、CPU11に内蔵されたRAM11b中のモードRAMをインクリメントする（ステップS71）し、モードRAMの値を判定する（ステップS72、S73）。

【0056】ここで、モードRAMの値が“0”のときはノーマルモード（ステップS74）、モードRAMの値が“1”のときはセルフ撮影モード（ステップS75）、モードRAMの値が“2”のときはスポット測光モード（ステップS74）にそれぞれ設定する。

【0057】また、モードRAMの値が“3以上”ならば“0”にリセット（ステップS76）し、その後でノーマルモードに設定する（ステップS77）。次に、図8のサブルーチン「フォーカスレンズ駆動」を参照して、その処理動作を説明する。

【0058】初めに、被写体までの距離に対応する量だけ、フォーカスレンズを繰出すが、このレンズ繰出し量は、スイッチSLがオフしてから、被写体までの距離に対応したフォトインタラプタPI\_Lのパルス数により決定されているものとする。

【0059】先ず、カウンタをリセットし（ステップS81）、モータMLを駆動する（ステップS82）。そして、スイッチSLがオフになるのを待ってから（ステップS83）、フォトインタラプタPI\_Lが立下がる毎に（ステップS84）、CPU11内のカウンタをインクリメントする（ステップS85）。 50

【0060】そして、カウンタが目標値になったならば（ステップS86）、モータMLにブレーキを掛け（ステップS87）、処理を終了する。次に、このフォーカスレンズ駆動のプログラムのバグ修正の例を説明する。

【0061】なお、バグ修正手法としては、（1）特開平3-186927号公報で示されているCPUを使用する手法、及び（2）特開昭62-52635号公報に記載されているような手法がある。

【0062】上記（1）は、ROM変更用メモリをCPU内に含んでおり、変更したいROMアドレスと変更コードを対で変更用メモリに記憶し、ROMアドレスとプログラムカウンタの値が一致したらROMアドレスに代えて変更コードを出力する手法である。

【0063】また、上記（2）は、変更したいアドレスと、プログラムカウンタの値が一致したら、特定のメモリにジャンプして修正する手法である。上記（1）、

（2）の何れも、（1）に示されているように、EEPROMに変更したいデータを記憶しておき、上記メモリにパワーオンリセット直後に、転送することで擬似的にプログラムを修正することができる。

【0064】この発明では、上記（1）、（2）のどち

らの手法を利用しても良いことは言うまでもない。また、プログラム変更用データを記憶するメモリは不揮発性であれば良く、EEPROMでなくともフラッシュメモリやPROMでも良いことも言うまでもない。

【0065】図9は、図8に示された「フォーカスレンズ駆動」のプログラムバグ修正の一例である。図8で説明したフォーカスレンズ駆動は、例えば、スイッチS<sub>L</sub>が壊れていたり、機構部品が壊れていたりすることにより、モータを動かしてもフォーカスレンズが動かない場合は、いつまでたってもモータが止まらなくなってしまうというバグが製造開始後に発生する場合がある。

【0066】したがって、この場合の修正手法は修正データに対して、スイッチS<sub>L</sub>がオフになるまでの時間にリミッタを付している。すなわち、ステップS82とS83の間に、タイマスタート（ステップC1）と、さらにこのタイマが所定時間を超えたら異常処理に移行してモータM<sub>L</sub>を停止させる処理（ステップC2）を追加している。

【0067】ROM修正可能なCPUを使用したシステムを利用すると、標準的な仕様のカメラシステムをユーザの希望に合わせて変更することも可能である。以下、ユーザの希望に合せたモードのことをデリバティブモードと記すことにする。

【0068】図10を参照して、デリバティブモードの第1の例について説明する。図3で説明したリリース処理動作は、2ndリリーススイッチ22がオンになってからフォーカスレンズを駆動するので、リリースタイムラグが長くなるのを短縮したいというユーザの要求に応えるものである。

【0069】そこで、2ndリリース（ステップS23）の前にフォーカスレンズ駆動を行い（ステップC3）、2ndリリース後のフォーカスレンズ駆動をジャンプさせるようにカメラを改造することができる。

【0070】次に、図11を参照して、デリバティブモードの第2の例について説明する。この第2の例は、ストロボ常時発光モードを全く使用しないというユーザの要求により、ストロボモードRAMが“2”になることを禁止したものである。

【0071】すなわち、ステップS62に代えて、ストロボモードRAMの値が“2以上”であるか否かを判定する処理（ステップC4）を設けている。ここで、“0”、または“1”であればステップS63へ、“2以上”であればステップS66へ進む。

【0072】また、図12は、デリバティブモードの第3の例を示したものである。これは、セルフモードよりスポット測光モードの方を頻繁に使用するというユーザの要求に応じて、モードの切替わる順序を変更した例である。

【0073】すなわち、ステップS73に代えて、モードRAMの値が“2”であるか否かを判定する処理（ス

テップC5）を設けている。そして、“2”であればステップS75へ、それ以外の値はステップS74に進む。

【0074】さらに、図13は、デリバティブモードの第4の例を示したものである。これは、ストロボはオートモードしか使用しないというユーザの要求に応じて、ストロボモードスイッチ26の使用を禁止した例である。

【0075】すなわち、ステップS13の処理の後、ステップS15及びS16をジャンプして、ステップS17の処理を行うようにしている。以上の他にも、スイッチの意味づけを変更したり、シーケンスやタイミングを変更することも考えられる。

【0076】次に、設計に不具合があった場合、CPUのバージョンを変更したり、基板のバージョンを変更したりする場合の生産切換えを理想的に行う手法について説明する。

【0077】図14は、基板とCPUの変更及び組合せの例を示したものである。図14（a）は、生産初期の状態であり、CPU11<sub>1</sub>にはROM1が内蔵されており、基板1は図示しないスイッチパターンやEEPROMが実装されているボードあるいはフレキシブル基板であり、このEEPROMには基板1を示すデータが記憶されているものとする。

【0078】この場合、スイッチ29の各一端がCPU11<sub>1</sub>の内部プルアップ抵抗付きのポートに入力されており、各他端がグランドに接続されていることにより、スイッチ29がオンになると、CPU11<sub>1</sub>の対応ポートが“L”レベルになるので、それを検出する手法である。

【0079】通常はこの手法で問題はないが、カメラシステム等の電源が電池の場合は、消費電力をできるだけ抑える必要がある。例えば、スイッチ29の1つが不用意に押され易いスイッチであったりすると、カメラシステム等を筐等に入れた場合、誤ってスイッチオンになって電池を消耗してしまう不具合があったとする。

【0080】この場合の恒久的な対策としては、図15のフローチャートに示されるように、スイッチ29のグランド側をCPU11<sub>1</sub>のオープンドレインポートに接続した状態で、オープンドレインポートをオン（ステップC11）してスイッチ29の状態を読み（ステップS91）、その後でオープンドレインポートをオフ（ステップC12）することにより、スイッチ29の状態を読むときのみオープンドレインポートをオンようにすることが考えられる。

【0081】しかしながら、生産開始後に、このような不具合が発覚した場合は、部品手配も済んでおり、急に恒久的な対策することは困難であるので、CPUにROM修正機能がない場合には次のような手順で対策を行う。

【0082】(i) ROM1が内蔵されているCPUと、基板1との組合せにより生産を開始する（不具合は残っている）。

(ii) ROM2（図15の修正されたプログラムが入っているROM）が内蔵されているCPUと、基板2（図14cに示すパターンが修正された基板）とを設計変更し手配する。

【0083】(iii) ROM2が内蔵されているCPUを手配できたときでも、基板1が残った場合は図14

(b)に示すようにパターンカットし、基板1を手修正してROM2と組合せる。

【0084】(iv) ROM2が内蔵されているCPUと、基板2との組合せにより対策完了となる。

以上のように、ROMが内蔵されているCPUと基板との厳密なロット管理が必要となるが、現実には、上記(i)のように、不具合が残ったまま生産を開始しなければならない。

【0085】しかしながら、ROM修正機能をもったCPU11<sub>1</sub>を使用している場合は、以下の対応が可能となる。すなわち、不具合がわかるまでは上記(i)に従ってROM1が内蔵されているCPUと、基板1との組合せにより生産を開始するが、ここで、不具合が発生する場合を想定して、EEPROM14<sub>1</sub>には基板1であるという情報を記憶させておく。

【0086】不具合が発生したら、上記(iii)のように、すぐパターンを手修正して対策する（図14(b)参照）

この場合、まだROM1であるから、図15に示されるように、ROM修正データをEEPROMに設定するが、基板は手修正であるから、基板1-1という情報をEEPROM14<sub>1-1</sub>に入れておく。

【0087】この場合、図14(c)に示されるような、修正されたROM2を内蔵しているCPU11<sub>2</sub>を手配することができれば、当然ROM修正データはEEPROM14<sub>2</sub>に設定する必要はない。

【0088】そして、パターンが修正された基板を手配することができた場合には、図14(c)に示されるようになり、EEPROM14<sub>2</sub>には基板2である情報を記憶する。

【0089】この場合、ROM2を内蔵しているCPU11<sub>2</sub>が入手できていればROM修正データはEEPROM14<sub>2</sub>に書込む必要はないが、まだROM1の場合はやはりROM修正データをEEPROM14<sub>2</sub>に書込む。

【0090】この場合、図2に示す第1の外部装置31を外部通信コネクタ15に接続してROM修正データの書き込みを行う。図16は、第1の外部装置31によるROM修正データの書き込み判断のフローチャートである。

【0091】ここで、基板No.（ナンバ）は、すでに

前の工程で書込まれているものとする。まず、カメラシステムの外部通信コネクタ15に接続された第1の外部装置31により、まず基板No.とCPU11のROM No.を読む（ステップC21、C22）。

【0092】ここで、基板1の場合は対策できないので（ステップC23）、ROM修正データは書込まない。次いで、ROM1の場合は（ステップC24）ROM修正データをEEPROMに書込み（ステップC25）、ROM1でなければ（ROM2以上）、すでにROMが修正されているので、ROM修正データは書込まない。

【0093】以上のように、ROM修正可能なCPUを使用していれば、ROMを内蔵しているCPUや基板のロット管理が必要なく、しかも不具合が発覚したときに、すぐに対策を施すことが可能となる。

【0094】図14~16の例では、基板No.とCPUのROMバージョンとの関係によって、ROM修正データを選択する手法について示したが、工場出荷前にバグが発覚したとしても、基板を修正すること無くバグを修復することができる場合には、ROMバージョンのみによって、ROM修正データを選択するようにしても良い。

【0095】この場合の例を図22に示す。まず、CPU11のROMバージョンを読み（ステップC61）、このROMバージョンから修正が必要であるか否かをチェック（ステップC62）し、修正の必要がない場合にはそのまま処理を終了する。

【0096】しかるに、修正の必要がある場合には、そのROMバージョンに対応したROM修正データを選択（ステップC63）し、EEPROM14にそれを書き込む（ステップC64）。

【0097】図17は、デリバティブモードとバグ修正用のROM修正データの記憶手法の一例として、EEPROMのアドレスマップを示しているが、ここで示されているアドレスは一例であるので、このアドレスに限定される必要はない。

【0098】図17-(a)は、デリバティブモードもバグ修正も有る場合の例である。この例では、アドレス57<sup>H</sup>はバグ修正用データのバイト数を、アドレス58<sup>H</sup>はデリバティブ用修正データのバイト数を示している。

【0099】この例では、00<sup>H</sup>~57<sup>H</sup>は、カメラシステムの駒数や調整値を記憶する領域であり、59<sup>H</sup>以後はROM修正用のデータエリアとなっており、このROM修正用エリアの中は、先ずバグ修正エリア用に割当てられ、次にデリバティブ用エリアとなる。

【0100】ここでは、バグ修正用データのバイト数は、57<sup>H</sup>に示されている10<sup>H</sup>バイトであるので、59<sup>H</sup>~68<sup>H</sup>がバグ修正エリアとなる。そして、デリバティブ用データのバイト数は12<sup>H</sup>であるので、69<sup>H</sup>(59<sup>H</sup>+10<sup>H</sup>)~7A<sup>H</sup>がデリバティブエリアとなる。



【0101】図17(b)は、バグ修正が無く、デリバティブモードのみ有る場合の例であり、この場合は、バグ修正バイト数(アドレス $57^H$ )を0バイトに設定すれば良い。

【0102】また、上述した例には示されていないが、デリバティブモード用のデータがない場合は、デリバティブ用バイト数(アドレス $58^H$ )を0にすれば良い。図18は、バグ修正とデリバティブ用のROM修正データをCPU内にセットするサブルーチンの例である。

【0103】先ず、 $A_R$ にEEPROM14のROM修正エリアの先頭番地( $57^H$ )を代入する(ステップC31)。次に、Nにバグ修正バイト数とデリバティブ用バイト数を加えたものを代入する(ステップC32)。

【0104】図17(a)の場合では、 $10^H + 12^H = 22^H$ をNに代入する。そして、Kに $A_R + N$ ( $59^H + 22^H = 7B^H$ )を代入する(ステップC33)。

【0105】ここで、KはROM修正用データの最終アドレスを表す。この後、ROM修正用データの有無を判定(ステップC34)し、ROM修正用データが無い場合はそのまま本ルーチンを抜け、上記ROM修正用データが有る場合( $N \neq 0$ )は、データをCPU11内のROM修正メモリに設定してやる(ステップC35)。

【0106】図19は、リペア店等で、ユーザがデリバティブモードを標準的なモードに戻したい時に使用するプログラムである。この場合、リペア店では前述した第2の外部装置32を用いてEEPROM14にデータを書き込む。

【0107】この場合、チェッカ通信で、このサブルーチンをコールし、EEPROM14のデリバティブ用バイト数(アドレス $58$ )を“0”にセットすれば(ステップC41)、デリバティブ用データは無いものと判断される。

【0108】図21(a)は、デリバティブモードがクリアされた状態のメモリマップを示している。図20は、リペア店等に於いて、デリバティブモードのデータを追加する時のサブルーチンであるが、ここでも、図19と同様に、チェッカ通信を利用してEEPROM14にデータを書き込む。

【0109】すなわち、先ず、外部装置よりデリバティブデータの総バイト数(D)とD個のデリバティブ用データを受取る(ステップC51)。次に、すでに書込まれているデリバティブデータの数にDを加算し、Kに代入する(ステップC52)。

【0110】これにより、Kは新しいデリバティブデータ数になる。次いで、Kにバグ修正バイト数を加えてNに代入する(ステップC53)。このNは、総ROM修正エリアのバイト数となる。

【0111】そして、Nと所定値とを比較し(ステップC54)、Nが所定値以上の場合はEEPROM14の容量がオーバーしてしまうので警告を出す(ステップC5

5)。

【0112】一方、上記ステップC54にて、容量がオーバーしない場合は、新しいデリバティブ用バイト数をアドレス $58$ にセットし(ステップC56)、新しいデリバティブデータをEEPROM14に追加する(ステップC57)。

【0113】図21(b)は、追加された状態のメモリマップを示している。これらからもわかるように、デリバティブモードのクリア、或いは追加する場合にも、バグ修正用のエリアには、何ら手を加えることはないの

で、誤ってバグ修正用データが変更されて装置として誤動作をすることがない。

【0114】なお、この発明によれば、以下に示す(1)～(19)のような構成を実現することができる。

(1) ワンチップマイクロコンピュータ本体と、修正データを受信するためのデータ入力手段と、ROMバージョンと前記ワンチップマイクロコンピュータ本体による制御のためのプログラム内容とを記憶したマスクROMと、前記マスクROMに記憶されたプログラム内容を前記データ入力手段を介して受信した修正データに基づいて擬似的に変更する擬似的変更手段とを含んでなるワンチップマイクロコンピュータと、前記ワンチップマイクロコンピュータによって制御される制御対象と、前記ワンチップマイクロコンピュータの外部に配置されるもので、前記マスクROMに記憶されたプログラム内容を変更するための前記修正データと、前記ワンチップマイクロコンピュータが搭載される電気回路基板バージョンを示す基板ナンバーとが記憶されている不揮発性メモリと、前記マスクROMに記憶されたROMバージョンと前記不揮発性メモリに記憶された基板ナンバーとを読み出し、この読み出されたROMバージョンと基板ナンバーとに適合する修正データを前記不揮発性メモリに書込むための書込み装置とを具備することを特徴とするワンチップマイクロコンピュータシステム。

【0115】(2) 前記書込み装置は、不具合対策されたことを示す前記基板ナンバーと、前記不具合対策以前の基板に対応しているROMバージョンであることを検出した場合に、これに対応する修正データを前記不揮発性メモリに書込むことを特徴とする(1)に記載のワンチップマイクロコンピュータシステム。

【0116】(3) ワンチップマイクロコンピュータ本体と、修正データを受信するためのデータ入力手段と、ROMバージョンと前記ワンチップマイクロコンピュータ本体による制御のためのプログラム内容とを記憶したマスクROMと、前記マスクROMに記憶されたプログラム内容を前記データ入力手段を介して受信した修正データに基づいて擬似的に変更する擬似的変更手段とを含んでなるワンチップマイクロコンピュータと、前記ワンチップマイクロコンピュータの外部に配置されるも

ので、前記マスクROMに記憶されたプログラム内容を変更するための前記修正データが記憶されていると共に、さらに前記マスクROMに記憶されたROMバージョンに適合する修正データが書き込まれる不揮発性メモリとを具備することを特徴とするワンチップマイクロコンピュータシステム。

【0117】(4) 前記不揮発性メモリには前記ワンチップマイクロコンピュータが搭載される電気回路基板を示す基板ナンバーが記憶されていることを特徴とする(3)に記載のワンチップマイクロコンピュータシステム。

【0118】(5) 前記不揮発性メモリには前記ROMバージョンと前記基板ナンバーとに応じて修正データが書き込まれることを特徴とする(4)に記載のワンチップマイクロコンピュータシステム。

【0119】(6) マスクROMに記憶されたプログラム内容を擬似的に変更可能な機能を有するワンチップマイクロコンピュータと、前記ワンチップマイクロコンピュータに接続され、前記マスクROMに記憶されたプログラム内容を擬似的に修正するための修正データを少なくとも記憶可能な電氣的に書き換え可能な不揮発性メモリとを有するシステムにおいて、前記ワンチップマイクロコンピュータ内にあって、前記マスクROMの種類を示すROMバージョンデータを記憶した記憶領域と、前記不揮発性メモリ内にあって、前記システム内の回路の種類を示す識別データを記憶した記憶領域と、該システムの外部から前記修正データを入力するための接続端子とを具備し、前記ROMバージョンデータと前記識別データとの組み合わせに基づいて適正な修正データが前記接続端子から入力されて前記不揮発性メモリに記憶されることを特徴とするシステム。

【0120】(7) 前記システムは前記接続端子を介して外部装置に接続可能であり、この外部装置は複数の修正データを記憶し、前記ROMバージョンデータと前記識別データとに基づいて前記複数の修正データから適切な修正データを選択して前記システムに入力することを特徴とする(6)に記載のシステム。

【0121】(8) マスクROMに記憶されたプログラム内容を擬似的に変更可能な機能を有すると共に、前記マスクROMの種類を示すROMバージョンデータを記憶しているワンチップマイクロコンピュータと、前記ワンチップマイクロコンピュータに接続され、前記マスクROMに記憶されたプログラム内容を擬似的に修正するための修正データを記憶可能であると共に、ROMバージョンデータに対応した修正データを記憶可能な記憶領域を有する電氣的に書き換え可能な有する不揮発性メモリと、前記ワンチップマイクロコンピュータもしくは前記不揮発性メモリに前記修正データを外部から入力するための接続端子とを具備することを特徴とするワンチップマイクロコンピュータシステム。

【0122】(9) 前記システムは前記接続端子を介して外部装置に接続可能であり、この外部装置は複数の修正データを記憶し、前記ROMバージョンデータに基づいて前記複数の修正データから適切な修正データを選択して前記システムに入力することを特徴とする(8)に記載のワンチップマイクロコンピュータシステム。

【0123】(10) マスクROMに記憶されたプログラム内容を擬似的に変更可能な機能を有すると共に、前記マスクROMの種類を示すROMバージョンデータを記憶しているワンチップマイクロコンピュータと、前記マスクROMに記憶されたプログラム内容を擬似的に修正するための修正データを記憶可能であると共に、前記ROMバージョンデータに対応した修正データを記憶可能な記憶領域を有する電氣的に書き換え可能な不揮発性メモリと、前記ワンチップマイクロコンピュータもしくは前記不揮発性メモリに前記修正データを入力するための接続端子とを具備することを特徴とするワンチップマイクロコンピュータシステム。

【0124】(11) 前記システムは前記接続端子を介して外部装置に接続可能であり、この外部装置は複数の修正データを記憶し、前記ROMバージョンデータに基づいて前記複数の修正データから適切な修正データを選択して前記システムに入力することを特徴とする(10)に記載のワンチップマイクロコンピュータシステム。

【0125】(12) カメラ本体と、前記カメラ本体を制御するために、マスクROMに記憶されたプログラム内容を擬似的に変更可能な機能を有すると共に、前記マスクROMの種類を示すROMバージョンデータを記憶しているワンチップマイクロコンピュータと、前記マスクROMに記憶されたプログラム内容を擬似的に修正するための修正データを記憶可能であると共に、ROMバージョンデータに対応した修正データを記憶可能な記憶領域を有する電氣的に書き換え可能な有する不揮発性メモリと、前記ワンチップマイクロコンピュータもしくは前記不揮発性メモリに前記修正データを入力するための接続端子とを具備することを特徴とするワンチップマイクロコンピュータを有するカメラシステム。

【0126】(13) 前記カメラシステムは前記接続端子を介して外部装置に接続可能であり、この外部装置は複数の修正データを記憶し、前記ROMバージョンデータに基づいて前記複数の修正データから適切な修正データを選択して前記カメラシステムに入力することを特徴とする(12)に記載のワンチップマイクロコンピュータを有するカメラシステム。

【0127】(14) マスクROMに記憶されたプログラムを擬似的に変更可能なワンチップマイクロコンピュータを有するシステムにおいて、前記マスクROMに記憶されたROMバージョンを外部に出力する出力手段と、この出力手段によって出力された前記ROMバージョン

10

20

30

40

50

ョンに基づいて、前記マスクROMに記憶されたプログラムのうち擬似的に変更を行うプログラムを選択する選択手段と、前記選択手段によって選択されたプログラムを前記ワンチップマイクロコンピュータに接続された不揮発性メモリに書き込む書き込み手段とを具備することを特徴とするシステム。

【0128】(15) 前記不揮発性メモリは、前記擬似的に変更を行うプログラムの容量に関するデータを記憶するエリアを有することを特徴とする(14)に記載のシステム。

【0129】(16) 前記選択手段は前記ワンチップマイクロコンピュータが搭載される電気回路基板バージョンを示す基板ナンバーを入力する入力手段を有し、前記基板ナンバーと前記ROMバージョンとに基づいて、前記マスクROMに記憶されたプログラムのうち擬似的に変更を行うプログラムを選択することを特徴とする(14)に記載のシステム。

【0130】(17) マスクROMに記憶されたプログラムを該プログラムの実行時に変更可能なワンチップマイクロコンピュータを有するシステムにおいて、前記マスクROMに記憶されたROMバージョンを出力する出力手段と、前記出力手段によって出力された前記ROMバージョンに基づいて、前記マスクROMに記憶されたプログラムの修正が必要であるか否かを判定する判定手段と、前記判定手段により修正が必要な場合、修正プログラムを書き込む書き込み手段とを具備することを特徴とするシステム。

【0131】(18) 前記修正プログラムは前記ワンチップマイクロコンピュータに接続された不揮発性メモリに書き込まれることを(17)に記載のシステム。

(19) マスクROMに記憶されたプログラムを該プログラムの実行時に変更可能なワンチップマイクロコンピュータを有するシステムの変更に用いる修正データを書き込む修正データ書き込み方法において、前記マスクROMに記憶されたROMバージョンを出力する出力ステップと、前記出力ステップによって出力された前記ROMバージョンに基づいて、前記マスクROMに記憶されたプログラムの修正が必要であるか否かを判定する判定ステップと、前記判定ステップにより修正が必要な場合、修正プログラムを書き込む書き込みステップとを具備することを特徴とする修正データ書き込み方法。

【0132】

【発明の効果】以上のようにこの発明によれば、内蔵されるマスクROMが修正された後のマイクロコンピュータと修正前のマイクロコンピュータとを自動的に判別し、生産ラインやリペア店で混乱の生じることなく、管理が容易なワンチップマイクロコンピュータシステムを提供することができる。

【図面の簡単な説明】

【図1】この発明のワンチップマイクロコンピュータシ

ステムが適用されたカメラシステムを例にとったブロック構成図である。

【図2】図1のカメラシステムの全体の動作を説明するメインフローチャートである。

【図3】「リリース処理」の動作を説明するサブルーチンである。

【図4】第1乃至第4の割込み処理の動作を示すフローチャートである。

【図5】「チェッカ通信」の動作を説明するサブルーチンである。

【図6】「ストロボモード処理」の動作について説明するサブルーチンである。

【図7】「モード処理」の動作を説明するサブルーチンである。

【図8】「フォーカスレンズ駆動」の動作を説明するサブルーチンである。

【図9】図8のサブルーチン「フォーカスレンズ駆動」のプログラムバグ修正の一例を示した図である。

【図10】デリバティブモードの第1の例について説明するもので、図3のサブルーチン「リリース処理」のプログラムバグ修正の一例を示した図である。

【図11】デリバティブモードの第2の例について説明するもので、図6のサブルーチン「ストロボモード処理」のプログラムバグ修正の一例を示した図である。

【図12】デリバティブモードの第3の例について説明するもので、図7のサブルーチン「モード処理」のプログラムバグ修正の一例を示した図である。

【図13】デリバティブモードの第4の例について説明するもので、図2のフローチャートのプログラムバグ修正の一例を示した図である。

【図14】基板とCPUの変更及び組合せをの例を示した図である。

【図15】図14の基板とCPUの組合せに不具合が生じた場合の恒久対策による処理動作を説明するフローチャートである。

【図16】ROM修正データの書き込み判断のフローチャートである。

【図17】デリバティブモードとバグ修正用のROM修正データの記憶方法の一例で、EEPROMのアドレスマップを示した図である。

【図18】バグ修正とデリバティブ用のROM修正データをCPU内にセットするサブルーチンである。

【図19】リペア店等で、ユーザがデリバティブモードを標準的なモードに戻したい時に使用するサブルーチンである。

【図20】リペア店等に於いて、デリバティブモードのデータを追加する時のサブルーチンである。

【図21】(a)は、デリバティブモードがクリアされた状態のメモリマップを示した図、(b)は追加された状態のメモリマップを示した図である。

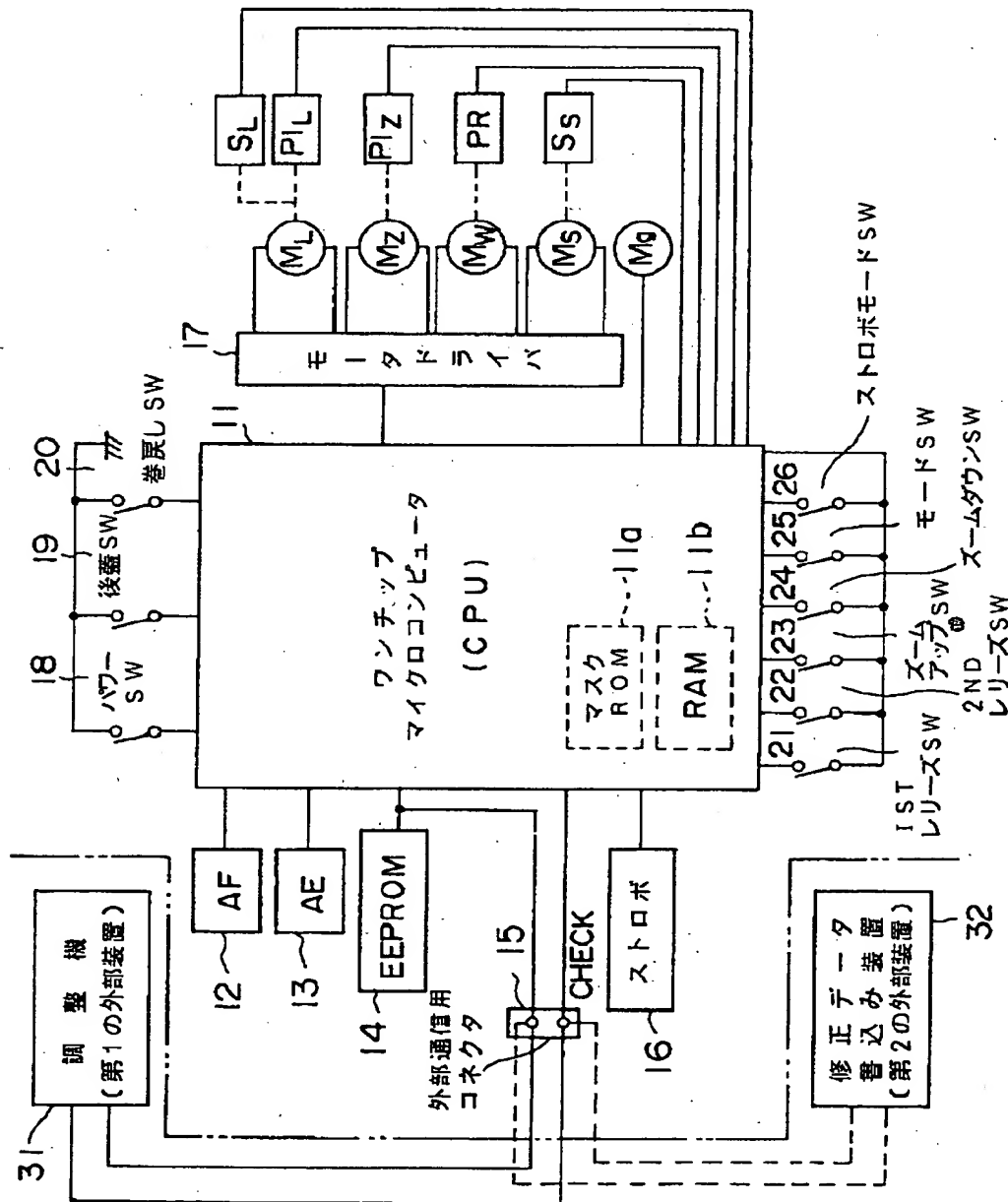
【図22】ROM修正データの書き込み判断の他の例を示すフローチャートである。

【符号の説明】

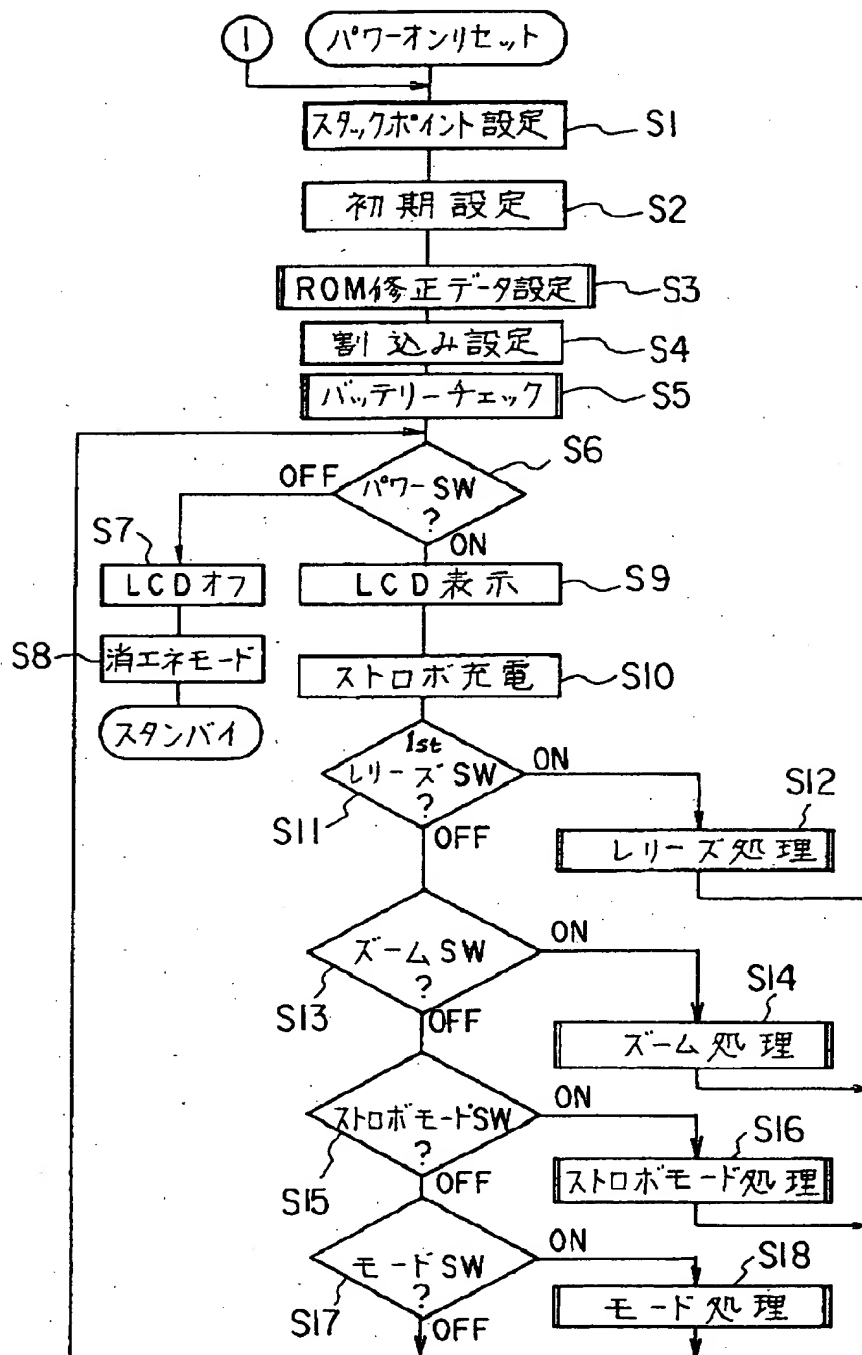
11…ワンチップマイクロコンピュータ (CPU)、12…AF回路、13…AE回路、14…EEPROM、15…外部通信用コネクタ、16…ストロボ、17…モータドライバ、18…パワースイッチ、19…後蓋スイッチ、20…巻戻しスイッチ、21…ファースト (1st) リリーススイッチ、22…セカンド (2nd) リリーススイッチ、23…ズームアップスイッチ、24…ズームダウンスイッチ、25…モードスイッチ、26…ストロボモードスイッチ、Mg…マグネット、M<sub>L</sub>…フォーカスレンズ駆動モータ、M<sub>S</sub>…シャッタ駆動モータ、M<sub>W</sub>…巻上げ巻戻しモータ、M<sub>Z</sub>…ズームレンズ駆動モータ、PIL、PIZ…フォトインタラプタ、PR…フォトリフレクタ、SL…フォーカスレンズ初期位置検出スイッチ、SS…シャッタ初期位置検出スイッチ。

ーズスイッチ、23…ズームアップスイッチ、24…ズームダウンスイッチ、25…モードスイッチ、26…ストロボモードスイッチ、Mg…マグネット、M<sub>L</sub>…フォーカスレンズ駆動モータ、M<sub>S</sub>…シャッタ駆動モータ、M<sub>W</sub>…巻上げ巻戻しモータ、M<sub>Z</sub>…ズームレンズ駆動モータ、PIL、PIZ…フォトインタラプタ、PR…フォトリフレクタ、SL…フォーカスレンズ初期位置検出スイッチ、SS…シャッタ初期位置検出スイッチ。

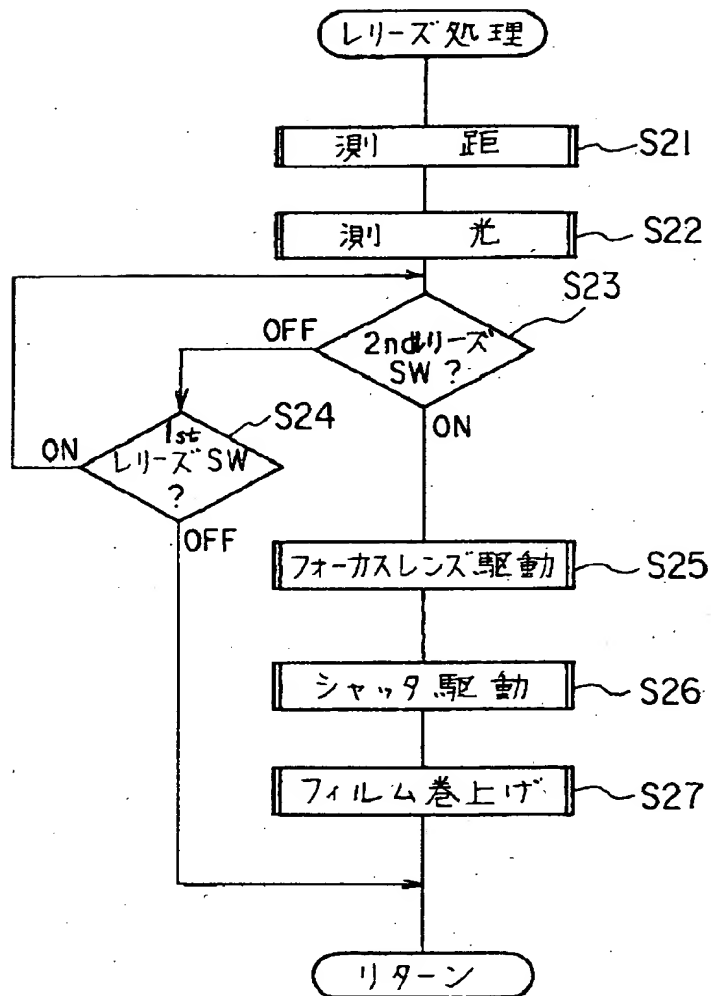
【図1】



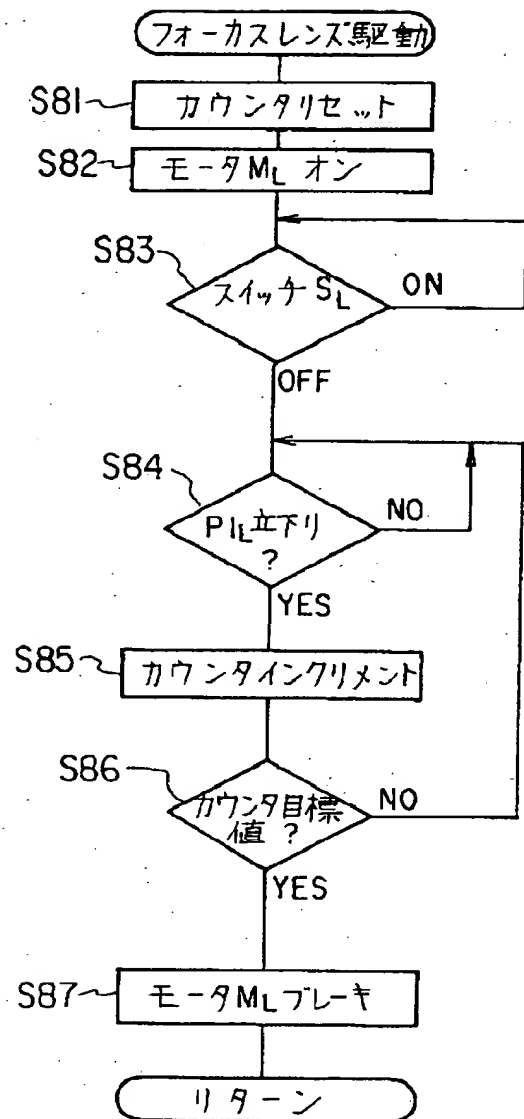
【図2】



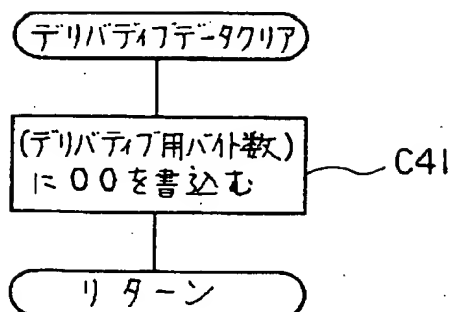
【図3】



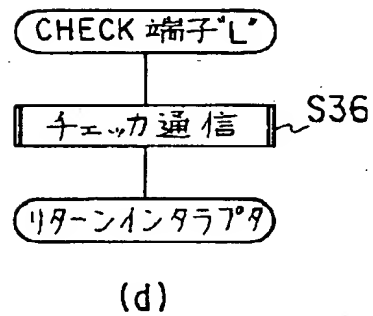
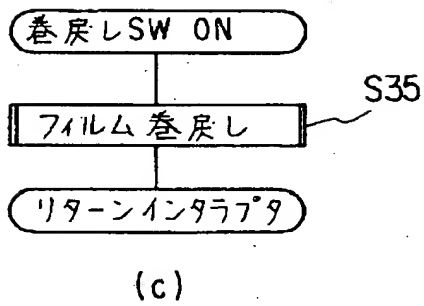
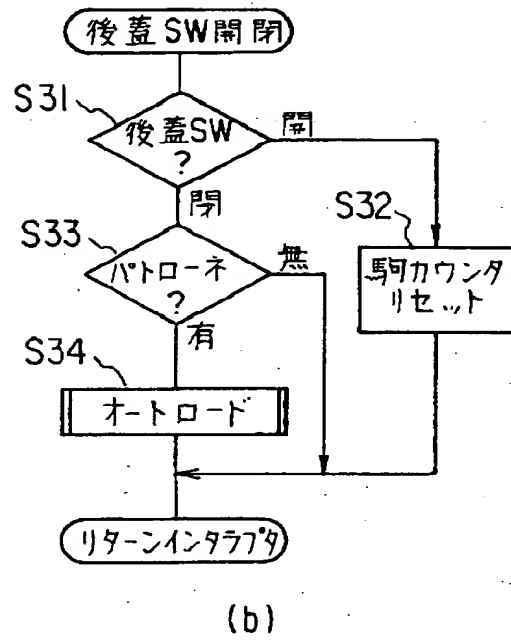
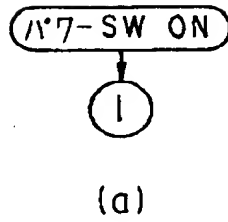
【図8】



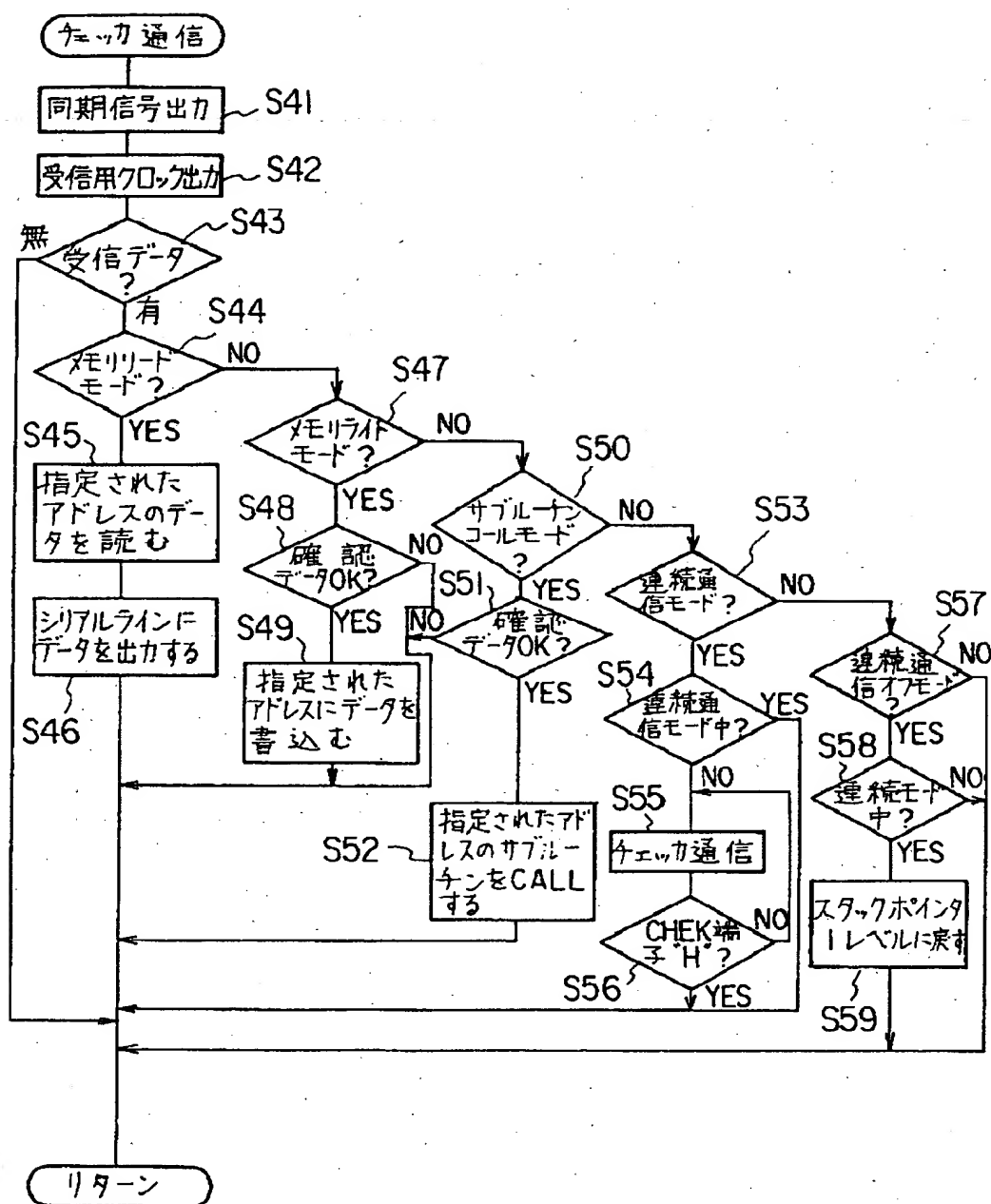
【図19】



【図 4】

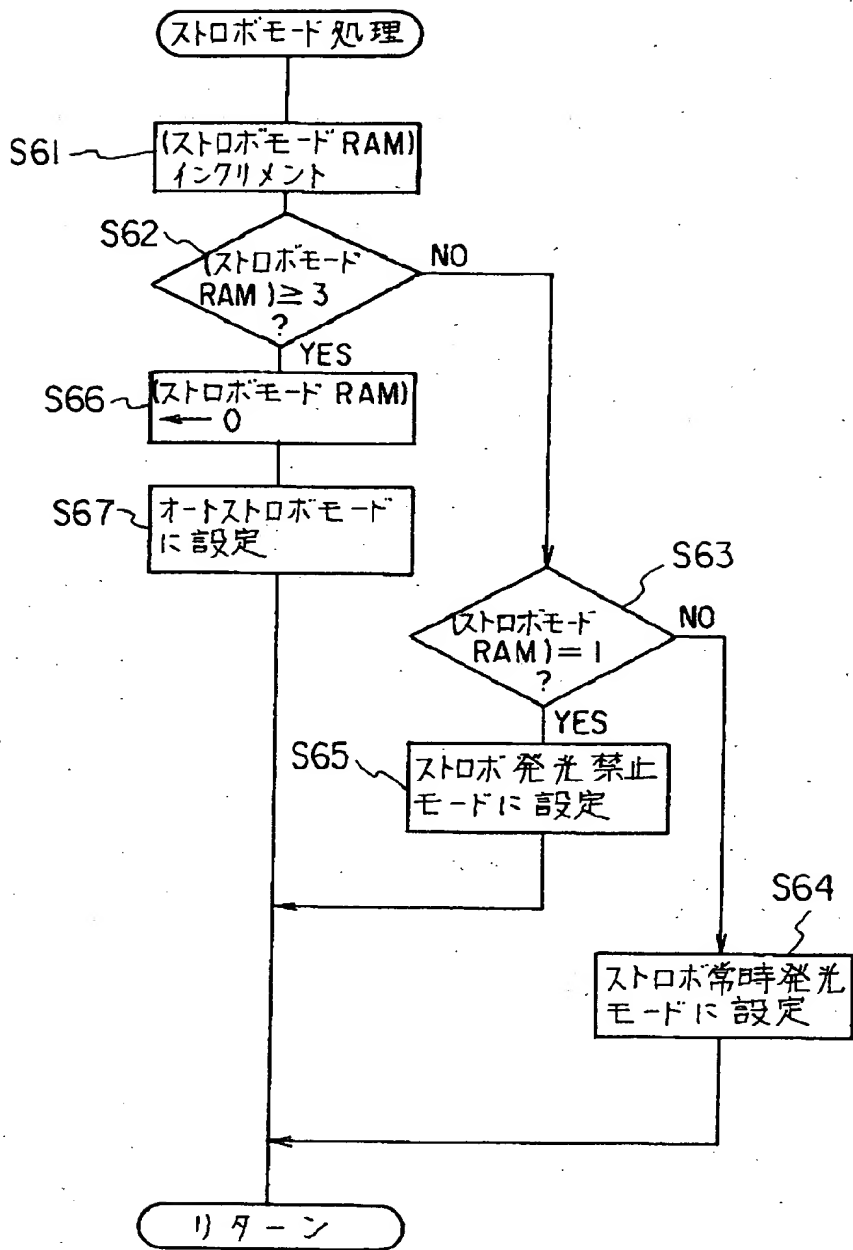


【図 5】

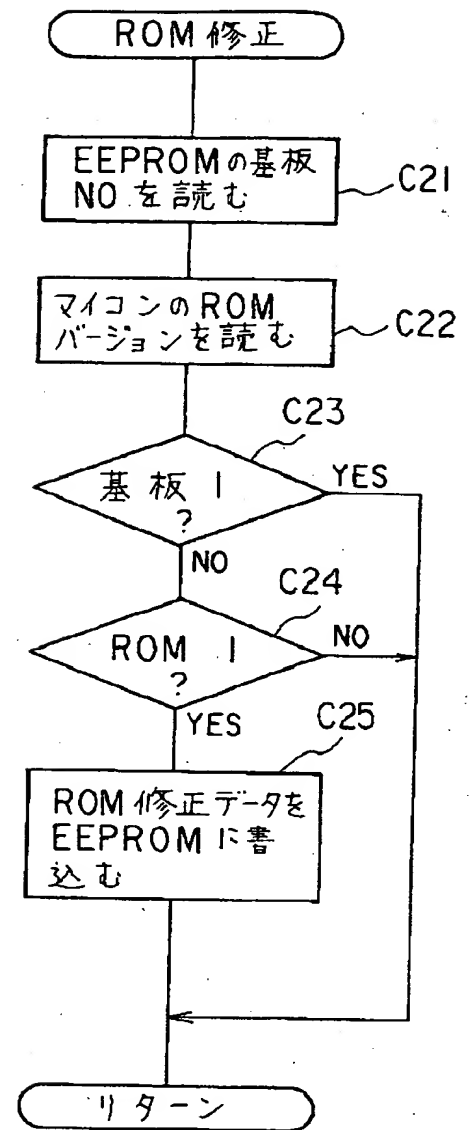




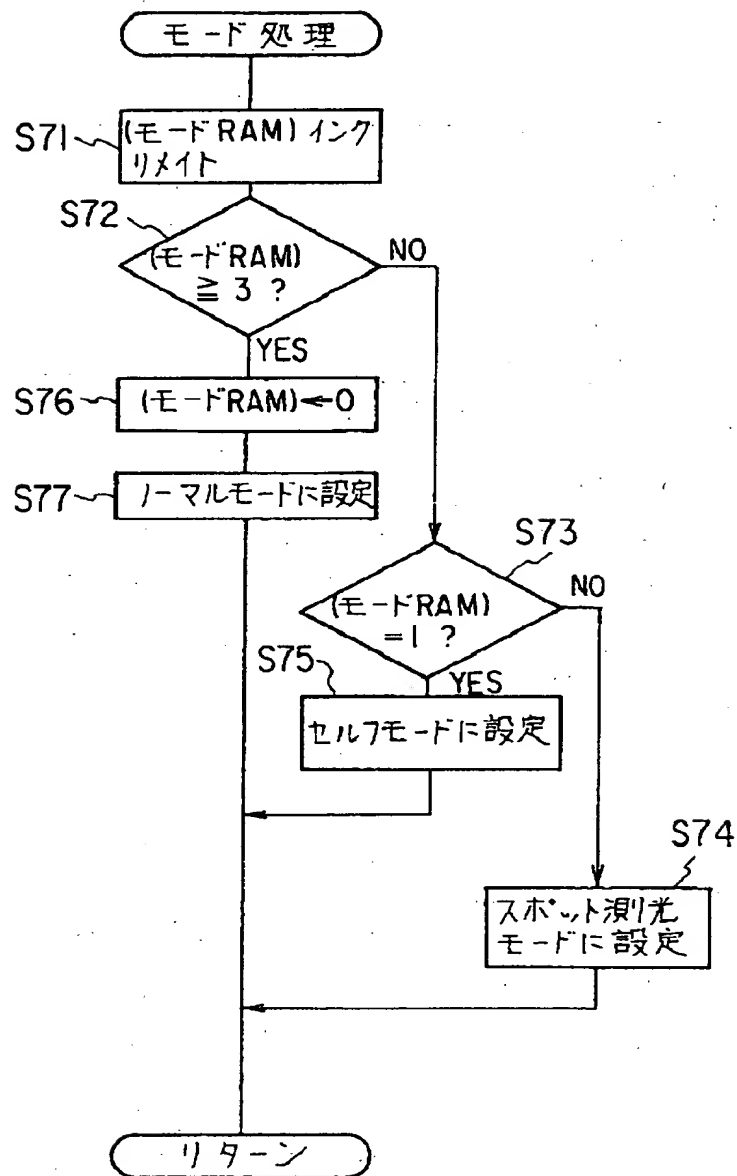
【図6】



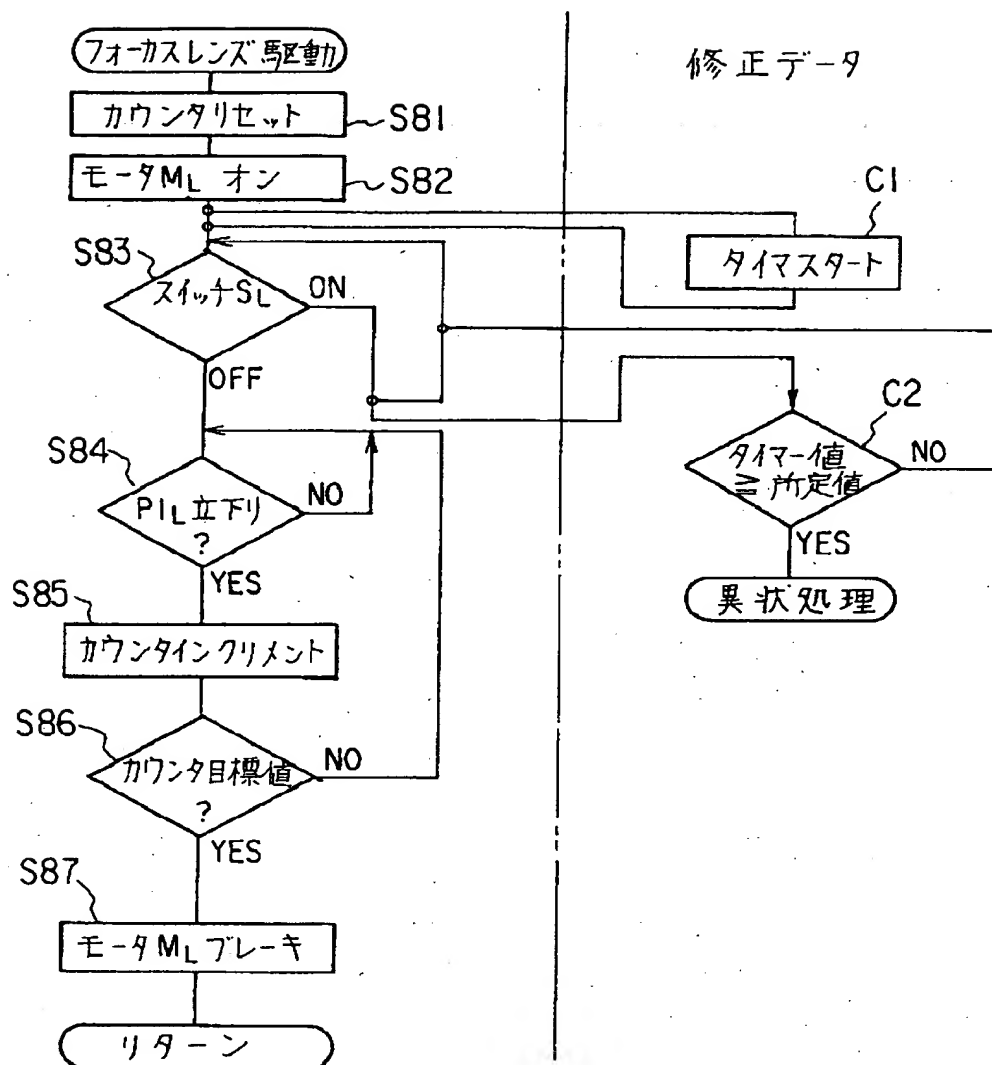
【図16】



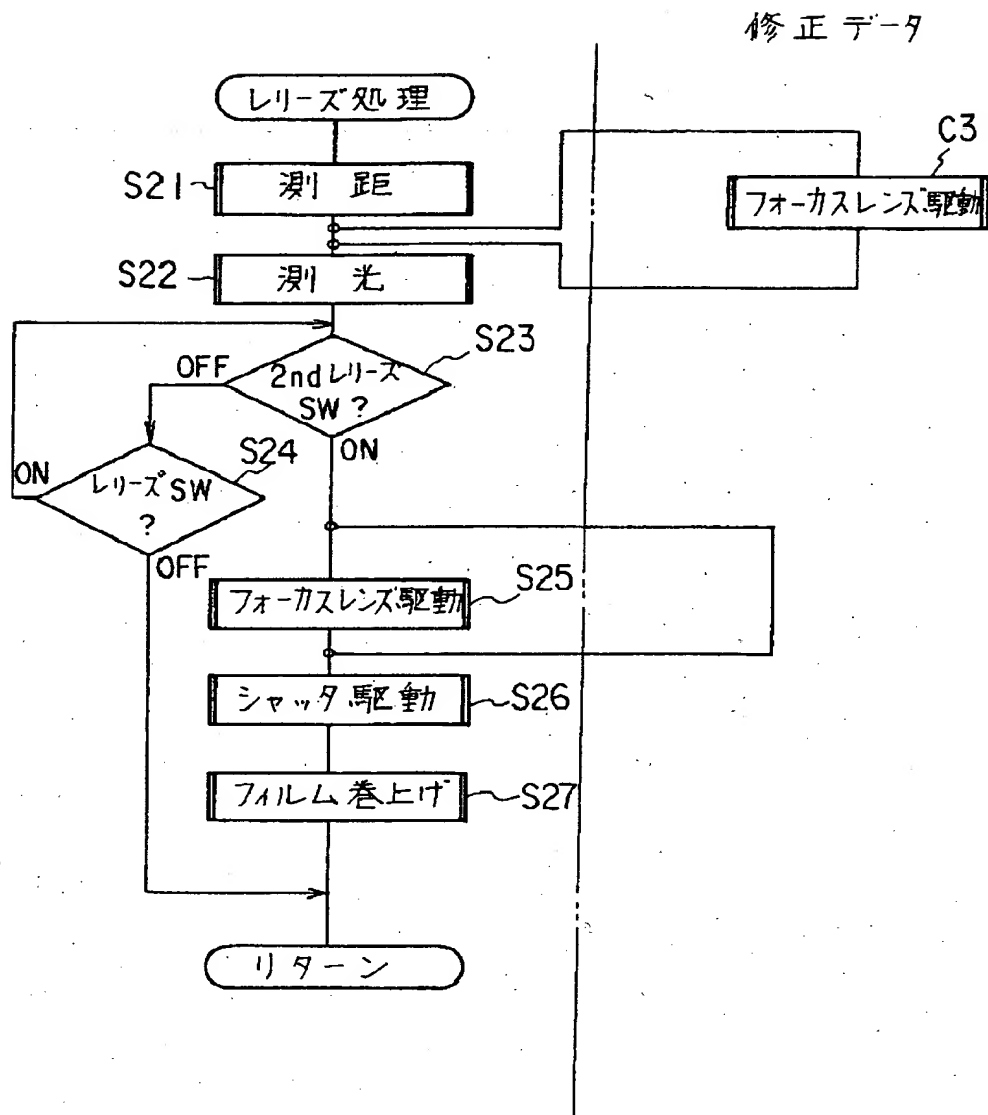
【図7】



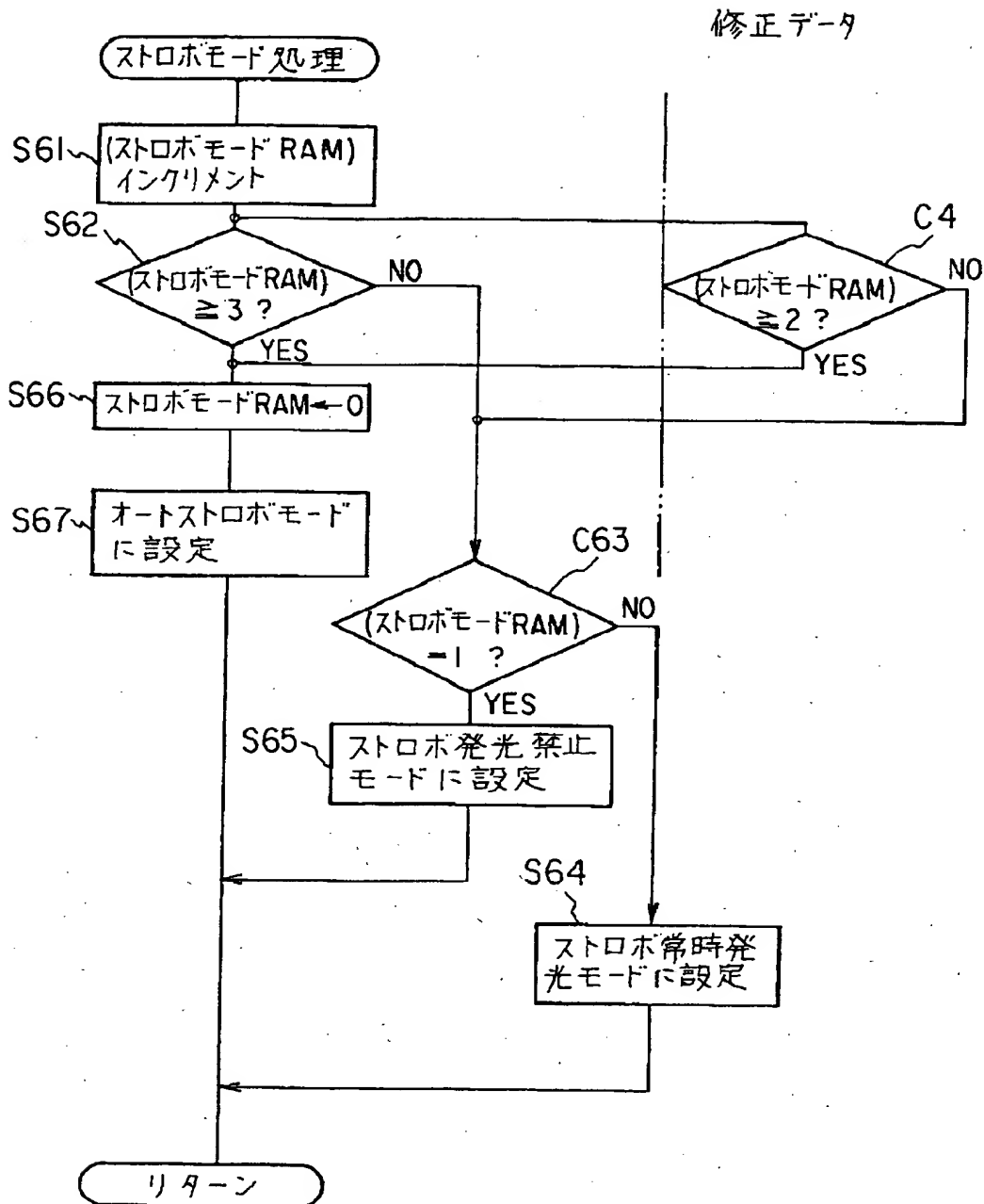
【図 9】



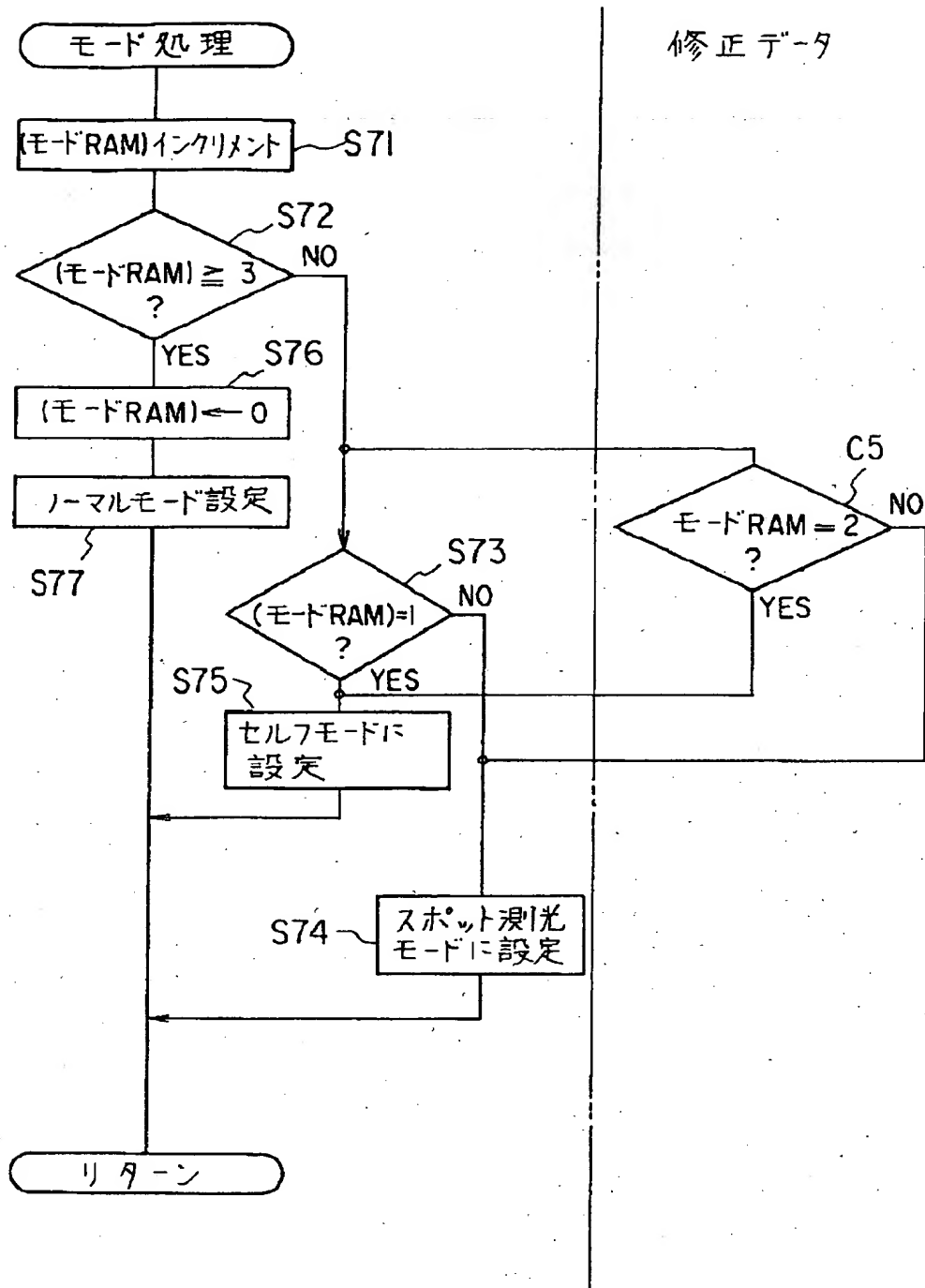
【図10】



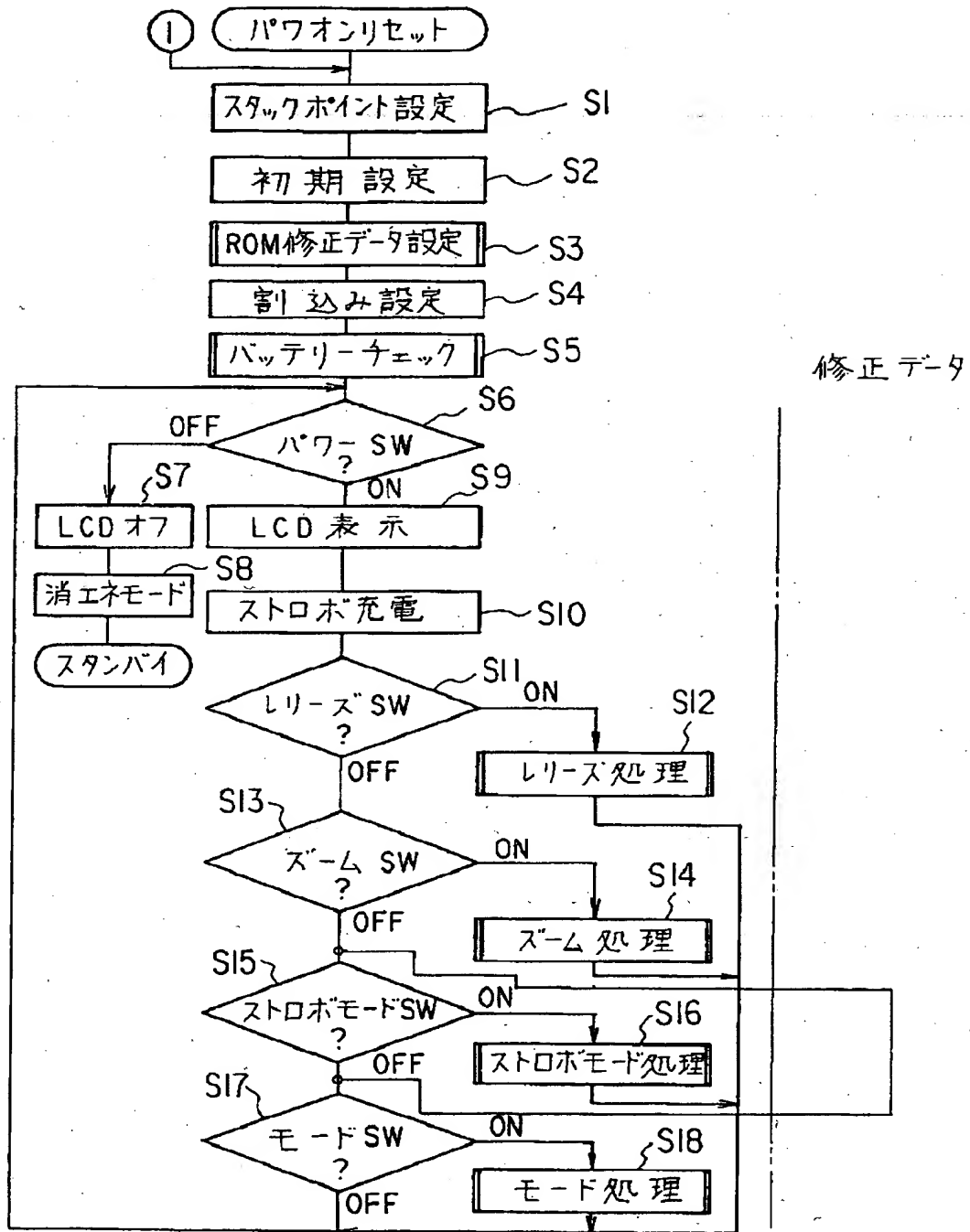
【図11】



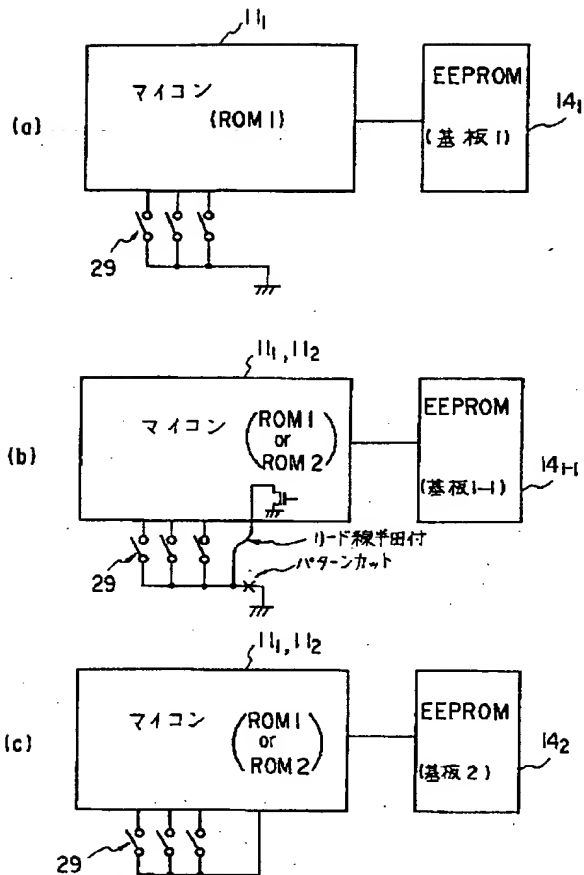
【図12】



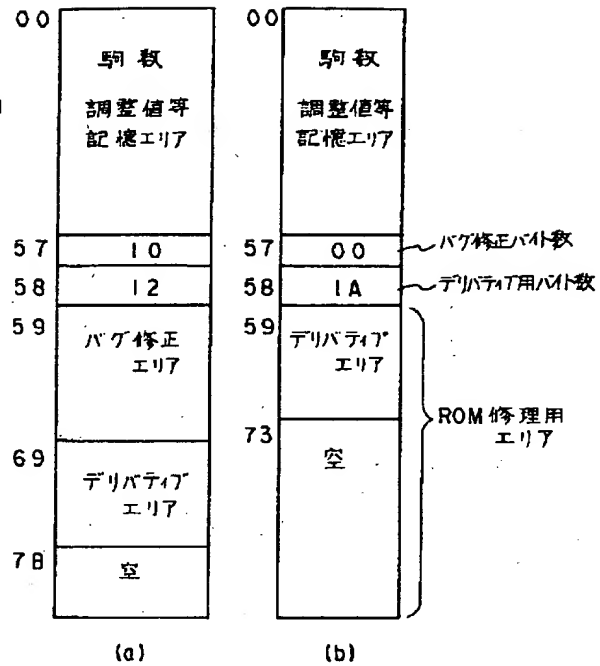
【図13】



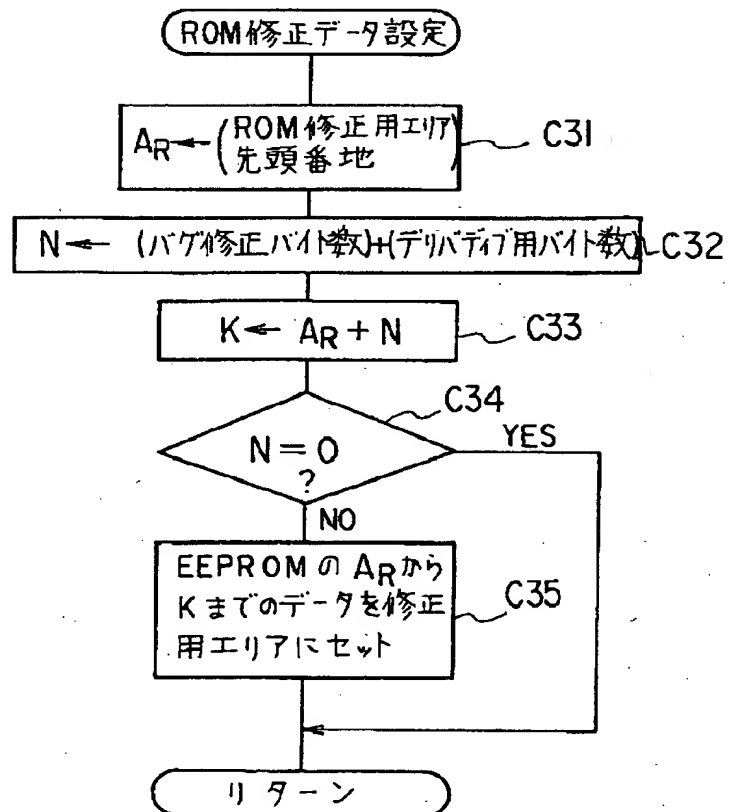
【図14】



【図17】

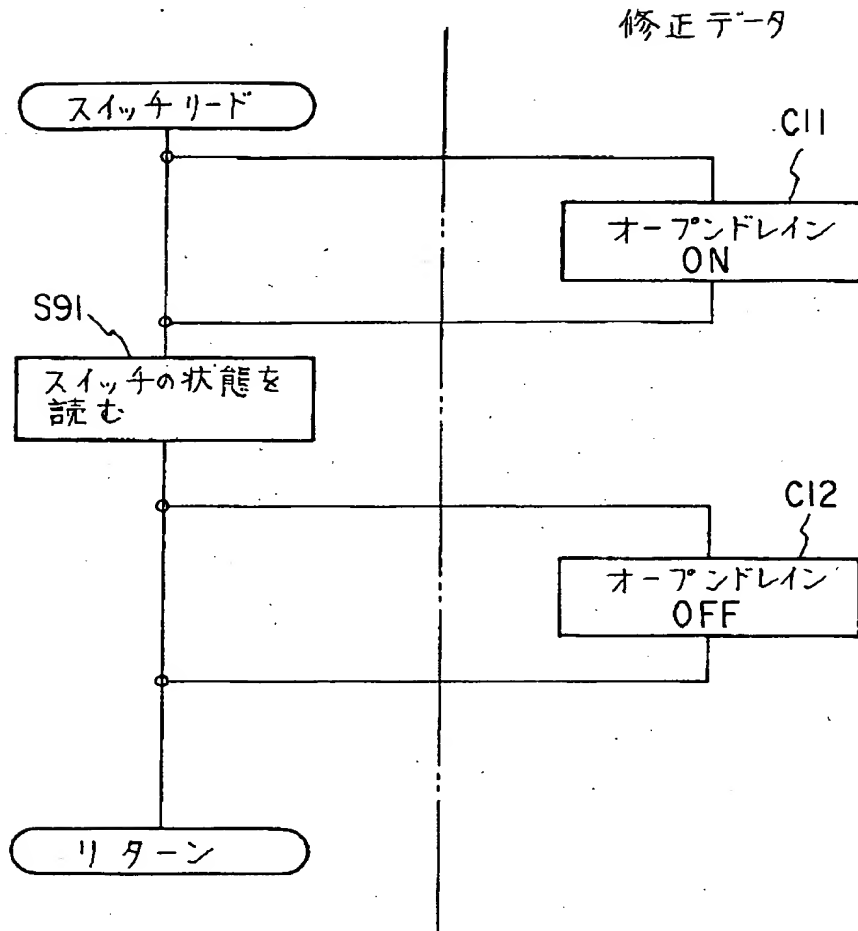


【図18】

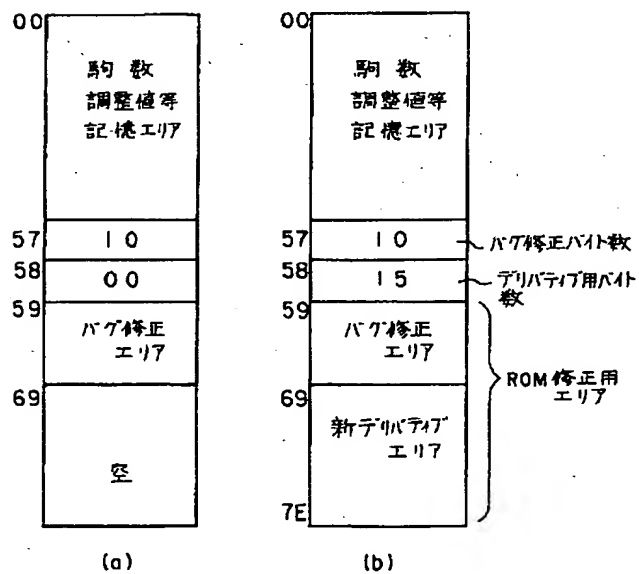




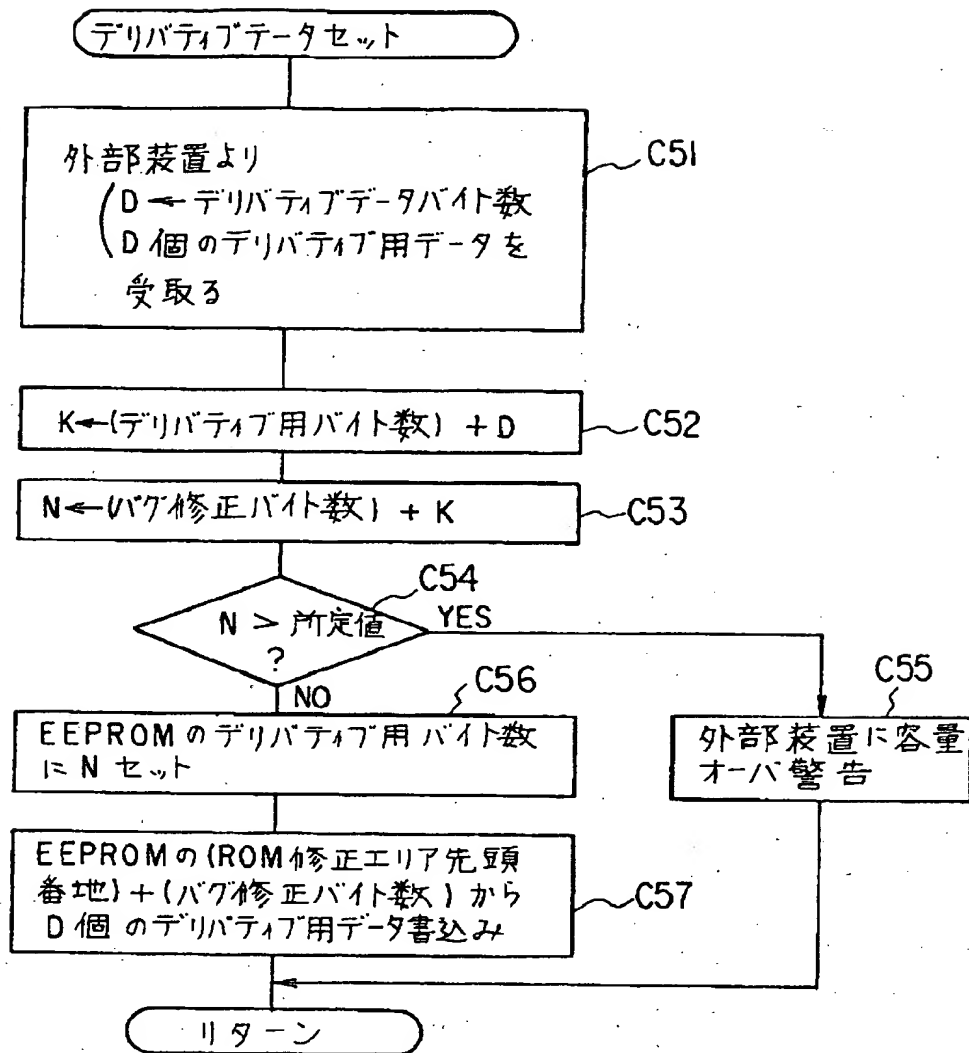
【図15】



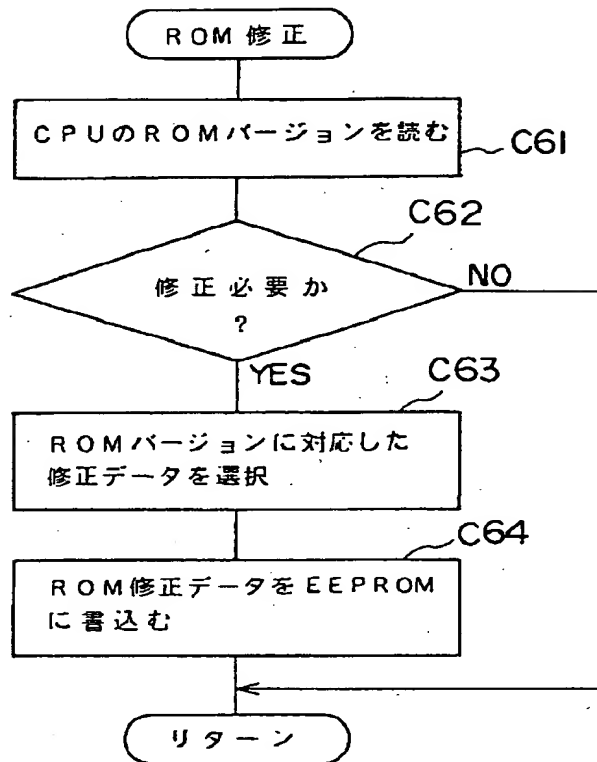
【図21】



【図20】



【図 22】



## フロントページの続き

(72)発明者 奥村 洋一郎  
東京都渋谷区幡ヶ谷 2丁目43番 2号 オリ  
ンパス光学工業株式会社内

(72)発明者 原 登  
東京都渋谷区幡ヶ谷 2丁目43番 2号 オリ  
ンパス光学工業株式会社内

(72)発明者 小林 一任  
東京都渋谷区幡ヶ谷 2丁目43番 2号 オリ  
ンパス光学工業株式会社内

(72)発明者 藤林 謙治  
東京都渋谷区幡ヶ谷 2丁目43番 2号 オリ  
ンパス光学工業株式会社内

(72)発明者 斉藤 裕一  
東京都渋谷区幡ヶ谷 2丁目43番 2号 オリ  
ンパス光学工業株式会社内

(72)発明者 川村 正二  
東京都渋谷区幡ヶ谷 2丁目43番 2号 オリ  
ンパス光学工業株式会社内